2015-06

# Scalable mobile ad hoc network (MANET) to enhance situational awareness in distributed small unit operations

Driesslein, Jonathan Clarke

Monterey, California: Naval Postgraduate School


# NAVAL POSTGRADUATE SCHOOL

**MONTEREY, CALIFORNIA**

# THESIS

**SCALABLE MOBILE AD HOC NETWORK (MANET) TO ENHANCE SITUATIONAL AWARENESS IN DISTRIBUTED SMALL UNIT OPERATIONS**

by

Jonathan Clarke Driesslein

June 2015

Thesis Advisor: James Calusdian
Co-Advisor: Zac Staples

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

# REPORT DOCUMENTATION PAGE

*Form Approved OMB No. 0704–0188*



| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE** June 2015 | **3. REPORT TYPE AND DATES COVERED** Master's Thesis |
|---|---|---|

**4. TITLE AND SUBTITLE**
SCALABLE MOBILE AD HOC NETWORK (MANET) TO ENHANCE SITUATIONAL AWARENESS IN DISTRIBUTED SMALL UNIT OPERATIONS

**5. FUNDING NUMBERS**

**6. AUTHOR(S)** Jonathan Clarke Driesslein

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Naval Postgraduate School
Monterey, CA 93943-5000

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
N/A

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES** The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**
Approved for public release; distribution is unlimited

**12b. DISTRIBUTION CODE**

**13. ABSTRACT (maximum 200 words)**

Platforms throughout the military and other government agencies (such as FEEMA and police departments) have become more networked; the last link in each network chain, however, has always been the individuals themselves. This structure requires a network that can process large amounts of data in order to provide the individuals with succinct and actionable information. Information, such as individual positions, weapons orientation, and friendly positions, serve to greatly enhance the situational awareness and improve the likelihood of mission success. The goal of this research is to use networking to improve the infantry's situational awareness.

The Robotic Operating System (ROS) is the foundation of a prototype network investigated in this thesis. It enables rapid prototyping of components and functionality through an open-source library with multi-language and multi-platform support. The network was constructed with software and hardware modules consisting of wearable sensors and various computational platforms. Future development will include linking the network to autonomous units and other assets with simplified controls.

The deliverable is a mobile ad-hoc network (MANET) with hardware designed to be operational for infantry squads and software designed to deliver contextual situational awareness to all of its members. The data distribution is handled through a brokered publish and subscribe network implemented via ROS.

**14. SUBJECT TERMS**
data dissemination, situational awareness, decision making, network-centric warfare, networking, mobile ad hoc network (manet), raspberry pi, robotic operation system (ros), arduino

**15. NUMBER OF PAGES** 123

**16. PRICE CODE**

| **17. SECURITY CLASSIFICATION OF REPORT** | **18. SECURITY CLASSIFICATION OF THIS PAGE** | **19. SECURITY CLASSIFICATION OF ABSTRACT** | **20. LIMITATION OF ABSTRACT** |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | UU |

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

# SCALABLE MOBILE AD HOC NETWORK (MANET) TO ENHANCE SITUATIONAL AWARENESS IN DISTRIBUTED SMALL UNIT OPERATIONS

Jonathan Clarke Driesslein
Ensign, United States Navy
B.S., United States Naval Academy, 2014

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**

from the

**NAVAL POSTGRADUATE SCHOOL**
**June 2015**

Author: Jonathan Clarke Driesslein

Approved by: James Calusdian, Ph.D.
Thesis Advisor

CDR Zac Staples, USN
Co-Advisor

R. Clark Robertson
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Platforms throughout the military and other government agencies (such as FEEMA and police departments) have become more networked; the last link in each network chain, however, has always been the individuals themselves. This structure requires a network that can process large amounts of data in order to provide the individuals with succinct and actionable information. Information, such as individual positions, weapons orientation, and friendly positions, serve to greatly enhance the situational awareness and improve the likelihood of mission success. The goal of this research is to use networking to improve the infantry's situational awareness.

The Robotic Operating System (ROS) is the foundation of a prototype network investigated in this thesis. It enables rapid prototyping of components and functionality through an open-source library with multi-language and multi-platform support. The network was constructed with software and hardware modules consisting of wearable sensors and various computational platforms. Future development will include linking the network to autonomous units and other assets with simplified controls.

The deliverable is a mobile ad-hoc network (MANET) with hardware designed to be operational for infantry squads and software designed to deliver contextual situational awareness to all of its members. The data distribution is handled through a brokered publish and subscribe network implemented via ROS.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

# LIST OF FIGURES

# EXECUTIVE SUMMARY

Dismounted infantry units need the ability to disseminate information amongst themselves and have a tether to their distant command post. In this thesis, a network capable of data dissemination for mobile dismounted infantry is investigated. A survey of technologies was completed that narrowed the choices for hardware and software to a handful of options. The best options of the choices left were prototyped and tested to create a managed publish and subscribe mobile ad-hoc network (MANET). The data disseminating network was tested with a program that detects friendly-fire situations based on physical positioning of squad members. The user is alerted through visual feedback of a potential friendly-fire situation. The product of this work is by no means a finished product and will need additional research to be fully actualized; however, a best fit and highly probable solution was suggested.

The military seeks to better disseminate tactical information to decision-makers. This flow of information needs to go in both directions in the chain of command. The unit leader on the ground needs information from the command post including orders, intelligence, terrain information, positions of friendly units, and other relevant information. Additionally, the overseeing authorities like to have relevant and prompt data from the ground. The faster that information can be delivered to command, the faster it can become actionable, medical support can be routed, and critical mission sensitive data can be relayed.

This problem is addressed in this thesis in a unique way—from the bottom-up. The goal is to provide individual Marines the ability to be linked into the network of information necessary for them to remain informed, coordinated, safe, and effective; and seeks to lift the fog of war through a flexible, scalable, upgradable, compatible network that can be integrated with larger systems such as vehicles, ships, and aircraft while compact enough to operate in a small form factor. Through multi-network integration a macro network that is accessible by outside groups was created, and a micro network in which units can pass useful and relevant information to each other was also created. Specifically, the micro level network of small, low-power, efficient devices networked

together using varying components, languages, and hardware was addressed. This type of network provides background for further component integration, improving the capabilities of the network on both the micro and macro scale. The system should be easy to use, reliable, intuitive, and possess the ability to be upgraded and expanded. As the smartphone has reimagined the way business works, so this network integration should redesign the way disasters are handled, wars are fought, and lives are saved.

A working prototype of a publish-and-subscribe network architecture in a MANET to disseminate information amongst a dismounted unit of Marines was implemented. The hope is that reversing the typical top-down build scheme by beginning at the bottom allows for more scalable network design at the macro level. If the problem can be solved at the lowest level, it is easier to implement and incorporate into the larger network. The test for this network's ability to deliver data amongst the nodes is a friendly-fire or fratricide detector for rifles. The network enables each node to know the location of every other node, which in turn is able to calculate whether conflicting geometries of fire exist and to warn the Marine holding the rifle. This fratricide detection program is an application chosen as a testbed to prove the concept that such a network can be built. The detection itself is not the aim of this thesis; only the network was focused on in this thesis.

Several design considerations were taken into account given the fratricide scenario. These factors affected choices in data refresh frequencies, baud rates, transmission mediums, and other networking overhead factors. The hardware and software choices were guided by the analysis, and the network was designed to be able to deliver the data to enough users quickly enough to be effective. The system of devices is overbuilt for this specific scenario in order to be adaptable to larger scale operations. Currently, the system is operational and able to deliver the data throughout the network to those who need it.

The network consists of several different devices to demonstrate interoperability and cross-platform integration. The main components are low-cost computers known as Raspberry Pis, while the lower computationally burdened devices are a popular brand of microcontroller development boards referred to as Arduino Unos. A Linux based desktop

computer was used in some testing to initially implement and build the network. Several connection mediums were also used including serial connections, wireless Xbee devices, and wired CAT5 connections.

The managed data broker network is implemented through the Robotic Operating System (ROS), software that was originally built to integrate varying sensors and actuators to be useable on a single robotic platform. That ability to integrate different devices and programing languages was used in this thesis to integrate the varying components within the network. The data within the network is organized by topic and is distributed by a broker program to those nodes within the network that are interested in those topics. This architecture made the network flexible and scalable. With data distribution managed by the network itself, the ability to redistribute it to new nodes is greatly improved. That ability to redistribute the data based on topic is a significant improvement for this implementation over client-server communications in a typical network.

With this network prototype, advancement of squad based technologies may be more easily achieved. Many of the technologies sought after need the ability to communicate amongst themselves through this network. It is far from the final iteration and is not field-ready, yet it serves as an important step forward in testing and prototyping. Future technologies now have a testbed to experiment and design new technologies to deliver to the field. As that experimentation continues, this network will grow and mature into a more field-ready product.

Future implementation for this network can be broken down into two categories: software improvements and hardware improvements. Each brings new functionality to the network. Software that better implements a more secure communications protocol helps make the network more field-ready. The current transmissions have no software based encryption but can currently be encrypted through the chosen communications medium such as the WPA encryption available through Wi-Fi routers. There can also be software based nodes that receive and analyze the data flowing through the network and make recommendations based on that data. The hardware improvements bring not only functionality but also the potential for new software implementations. An added camera

system to the nodes would provide more immediately actionable information to higher echelon leadership. This video feed can also then be passed through software that can identify faces within the video and compare them against databases to identify high value targets. Medical sensors could also be added to the system to allow for better monitoring and better reaction times for medical personnel. Perhaps most interesting is the possibility of integrating autonomous resource control into the network. As the number of robotic systems in the military increases, the ability to control them from the network will be an invaluable asset. Many questions need to be answered, including questions of interfacing and control for the individual Marine to manipulate the system.

With a working prototype of a managed publish and subscribe MANET, the possible implementations will only be discovered through further experimentation and analysis. The foundation for future research was laid in this thesis. Given the use of ROS as the data broker, the ability to integrate and collaborate is significantly easier. Concurrent and future research should be able to interface with the network and utilize the information available.

# ACKNOWLEDGMENTS

This thesis was accomplished only through the support and encouragement of my family, especially my fiancée, Catherine, who has supported me through this entire process. Special acknowledgment goes to Dr. Bishop and Dr. Piepmeier for encouraging and enabling me to have this opportunity. To Caleb Khan, for his mentoring and support; to Chloe Woida and Camille Rogers from the Graduate Writing Center, for their dedication and support; and to Dr. James Calusdian, for his guidance and mentorship.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

> We must build forces that draw upon the revolutionary advances in technology of war…one that relies more heavily on stealth, precision weaponry, and information technologies.
>
> —George W. Bush

Today's technology permeates our daily lives. Our cellphones provide navigation, schedule organization, communication, Internet access, and other features that make our lives more efficient and our time more effective. Compared to the capabilities provided by the technology we carry as civilians, our soldiers in the field currently have technology equivalent to a handheld radio. Imagine going through your daily routine with only a radio—no computers, no Internet access, no Google. That is the operating environment our soldiers have in the field. In this thesis, an idea to deliver the same technological capabilities that civilians experience in daily life to our soldiers in the field is investigated.

Research is currently ongoing in the area of new network architectures that can operate without centralized control or pre-established infrastructure. Specifically, research in Mobile Ad Hoc networks (MANETs) seeks to implement a decentralized control scheme to allow for mobile networks to have infrastructure-independent operational capacities. There is also research on making network software that is independent of the physical hardware. How data flows through the physical network components is one focus of this research. As the physical control of the network becomes decentralized, the routing of data through that network must adapt in order to operate effectively. The research in these areas is highly focused on theoretical and simulated environments. None of these systems has been built or tested at the Naval Postgraduate School.

Technological systems that help to share common operating pictures and increase situational awareness currently exist for the military, but the scale is typically so large and the equipment so cumbersome that only large-scale platforms can utilize them. No practical connection between these large-scale systems and the infantryman exists. In this

thesis, providing the modern infantry squad with the networking structure necessary to implement enhanced capabilities is investigated. It is not currently a bridge from the infantry squad to the larger system but has the potential to be. The hardware and software necessary to run a network at the infantryman's scale is the subject of this thesis.

There are three driving ideas that frame the cause for these awareness enhancing networks in the military: network-centric warfare, situational awareness, and decision-making. These interrelated and codependent ideas drive the ethos of both the existing awareness enhancing networks as well as the prototype network investigated in this thesis. Network-centric warfare is a doctrine with quantifiable tenets that help to identify technologies that improve effectiveness in the battle space. Doctrinally, situational awareness has been preached by military leaders since Sun Tzu and continues to be a relevant topic to the military today. Perhaps most impactful are the decisions made on the battlefield, which the previous two ideas inform. These three philosophical ideas are discussed more thoroughly in Chapter II.

The thread that ties the principles introduced above together is communication. Without communication none of these concepts are effective. Good communication, drawn from the principles of network-centric warfare, is a prerequisite to good situational awareness and decision making. Again, more context is discussed in Chapter II.

The problem is that, although networking solutions exist on vehicles and ships to share data within the military, enhance situational awareness, and enable better decision making, these systems do not reach down to individuals on the ground. The challenge of a network that can connect infantrymen together is that it would operate in environments without infrastructure. There are no cell phone towers and potentially no satellite connections. All broadcasting power and connections are from device to device. This brings up questions of power, reliability, and energy. How do the devices carry enough energy? What broadcast methods are the least expensive? What devices can run this network? How is it powered? What is the best way to deliver the data? What network structures are best suited for this?

The main questions investigated in this thesis are related to network topologies, device interfacing, and the size and weight of devices running the network. These devices, while running the network, must also possess the ability to be expanded to enable future iterations of technological advancement. The solution needs to effectively enable communication and distribute information to all those who need it, whether in the squad or in command and control. Devices used must be able to operate with other devices already in place. Interoperability must replace proprietary technologies. Finally, the size and weight of the devices necessary to run the network must not overly burden an infantry Marine who already carries significant amounts of equipment.

The research done for this thesis was approached through analysis of different network topologies and software that provide efficiency, scalability, and reliability for data dissemination. Several different devices were used throughout the network to show interoperability both in hardware, software, and communications protocols. Several test scenarios were devised to investigate the network in realistic situations. These situations also guided the choices of hardware, software, and communication protocols towards devices that can survive in the battlefield.

The goal of this thesis is to propose and implement a working prototype network for dismounted infantry units. To do this there must be an infrastructure to connect the devices to each other and eventually to a larger network. A prototyped network to include devices and sensors carried by infantry as a solution to that infrastructure problem is described in this thesis and employs the use of an open-source software called Robotic Operating System (ROS) to uniquely implement a network architecture that delivers data using a publish and subscribe configuration.

This thesis effort is part of a larger concurrent Naval Postgraduate School project titled Reticle that seeks to bring mobile technologies to the battlespace. In order to deliver this prototype network, a scenario was implemented to test the network's ability to communicate and exchange data. The Reticle project, which seeks to detect friendly-fire situations within a squad of Marines, was chosen to run on the prototyped network. The program provides a squad of infantrymen with the ability to detect the potential of friendly-fire and prevent it by warning the Marines of the situation. The prototype

network is not built for the specific application of friendly-fire detection, but this scenario is used to demonstrate the network's abilities. This scenario involves a squad of infantrymen and is informed through discussions with Marines but is not limited in application only to the Marines. The friendly-fire scenario, though simple in concept, proves to be complicated in design. Complications associated with the integration of this program's scenario to the network were characteristic of complications that had to be considered as a whole. Throughout the design phase, it had to be kept in perspective that the network was built for a scope much larger than the scenario it was tested in. The prototype network is not designed to be ready for the field but is instead an academic testbed intended for further research.

More background on the theory and importance of communication, situational awareness, the role of command and control, and network-centric warfare is provided in Chapter II. Discussions also include current technologies that are networking the military's higher level assets together, how they generally work, and why they need to reach down to the lowest level. A technical overview of network topologies as well as other technologies that are necessary to understand before continuing into the prototype network is provided in Chapter III. The layout and the software used to create the network are outlined in Chapter IV. Test conditions are parameterized and outlined as well as explanations made about how the data flows throughout the network. The results both in terms of how well the project meets the research goals and the methodological results are analyzed in Chapter V. Finally, the future growth of the project and suggest areas of continued study are discussed in Chapter VI.

# II. BACKGROUND

> War is the realm of uncertainty; three quarters of the factors on which action in war is based are wrapped in a fog of greater or lesser uncertainty. A sensitive and discriminating judgment is called for; a skilled intelligence to scent out the truth.
>
> —Carl Von Clausewitz

Background to the work done in this thesis and the importance of the infantry based network are provided in this chapter. The focus is on network-centric warfare, situational awareness, and decision making control. Discussion continues into networking technologies currently being used by the military to address some of these challenges. This sets the stage for the discussion of the construction of a network architecture at the infantry level. Throughout the discussion of these background ideas and technologies, it is important to keep in mind the common thread that they all rely on—*communication*.

## A. NETWORK-CENTRIC WARFARE

General Stanley McChrystal says of his years commanding forces both in Iraq and Afghanistan,

> It became clear to me and to many others that to defeat a networked enemy we had to become a network ourselves. We had to figure out a way to retain our traditional capabilities of professionalism, technology, and, when needed, overwhelming force, while achieving levels of knowledge, speed, precision, and unity of effort *that only a network could provide*. [Emphasis added] [1]

General McChrystal continued to speak about the disparity between United States military's networking ability and that of an insurgent enemy. Proprietary and complicated systems hindered U.S. military communication while cellphones and email enabled the enemy's. "It will require an open architecture that will allow further plug-and-play development in the future as our network grows and matures," says Army Vice Chief of Staff General Peter Chiarelli [2].

The need for the ability of a network to connect various components of the military was first formally addressed in the mid-1990s. The formulation of that need into

specific, measurable requirements has been labeled network-centric warfare (NCW). The pursuit of network-centric warfare increased after the second Gulf War, and interest in the subject grew even more after the recent cyber-attacks on American companies including Boeing, Sony, and various others. Several clarifying documents have been written to shape and define network-centric warfare. Most prominent is a report from the Department of Defense to Congress about network-centric warfare, in which the tenets of network-centric warfare are formulated [3].

1. A robustly networked force improves information sharing.
2. Information sharing and collaboration enhance the quality of information and shared situational awareness.
3. Shared situational awareness enables self-synchronization.
4. These, in turn, dramatically increase mission effectiveness.

NCW involves both: "The provision of vastly increased access to information at all echelons, and a redefinition of the relationships among participants in a mission and between commanders and subordinates" [4].

For the Department of Defense, mission effectiveness is always the end goal. Because of that, these tenets are heavily addressed in this thesis. The idea in Tenet 1 is taken much further than simply sharing information in terms of intelligence. The proposition is that data dissemination leads not only to situational awareness but eventually to better decision making data. Pieces of information taken individually do not always paint the whole picture. If sensors sent data reporting the temperature of the environment and the body temperatures of those in the squad, this may or may not mean anything to the leader; however, if a program were in place that contextually analyzed that data and reported that the squad's body temperatures were dropping and reported how much time they could remain exposed to the elements without suffering hypothermia—that would be useful, contextual information. That is the situational awareness of interest in this thesis research. Numbers and facts are only clutter without interpretation. Network-centric warfare, as interpreted here, includes analysis in conjunction with data sharing to create the shared situational awareness discussed in Tenet 2.

The ability that allows the tenets of network-centric warfare to be developed is communication. Without a communication infrastructure, the tenets of network-centric warfare are not achievable. Though each tenet builds on the one previous, there is an assumption of an infrastructure preceding Tenet 1—a networked communications infrastructure. This unacknowledged prerequisite involves not only a physical medium through which to communicate but also the sensors necessary to collect the data that will be communicated. That prerequisite infrastructure is sought after in this research. The goal is to have a sensor communications network integrated with the situationally aware interface. The networking necessary between the users is the same network that links the sensors. One network shares the raw data and the analyzed data at the same time. A prototype network that enhances both communication and data delivery assurance is the subject of this thesis. The National Defense Magazine, in their article entitled, "Army Under Pressure to bring broadband to the Battlefield," states that,

> Current battlefield networks are accessible by divisions, brigades and battalions. But smaller units remain digital orphans, even though they lead the day-to-day fighting in current wars. [2]

## B. SITUATIONAL AWARENESS

Carl Von Clausewitz, a well-respected author of war strategy and author of *On War*, coined the phrase the "Fog of War." This concept plays a significant role in large organizational operations, especially the military. The Fog of War is the uncertainty that occurs from not being able to access all the information on a battlefield. The concept of the Fog is that it obscures the view not only physically of what may be beyond a hill but also psychologically as to what strength the enemy forces may have. It particularly affects an attribute of the second tenant of network-centric warfare, situational awareness. Situational awareness (SA) and common operating picture (COP) share many similarities and are both discussed here. For simplicity they are referred to as situational awareness within this thesis. Dr. Dusseau, in a conference paper titled, "Designing User Friendly Situational Awareness Products," writes,

> Situational awareness…and Common Operation Picture are all terms used to describe getting real-time tactical information into the hands of the warfighter, whether that be a pilot in the cockpit of an aircraft or troops on

> the ground. This information has to be part of a complete and seamless pipeline of data that spans the breadth of pre-mission planning, mission rehearsal, and mission execution. *This ability to connect into the operating picture is especially critical in the modern era of rapid, long-range military deployments. This network centric warfare also requires an effective military communications network.* [emphasis added] [5]

Through an effective communications network, situational awareness combats the uncertainty caused by the Fog of War. Minimizing uncertainty increases the amount of available information and results in better decision making [6]. To have situational awareness, the right information must be delivered to the right people, at the right time, in the right way. It cannot be emphasized enough that situational awareness must be unilateral. In order to operate successfully, information must be commonly accessible, plans must be shared, changes must be disseminated, and decisions confirmed by all parties. This type of clarity prevents confusion, counterproductive actions, and unnecessary waste. The National Defense Magazine in their article, "U.S. troops loaded with technology, but can't harness the Power of the network," explains that in the recent decades of conflict it has become clear that, "U.S. soldiers have the most sophisticated weaponry and equipment in the world, and yet the enemy can outwit them simply because they have better means to receive and disseminate information" [1]. Within that article General Chiarelli was quoted saying that he has called for "a network that would allow soldiers to tap their laptop or smart phone keyboards and obtain the information they need, as well as pass around critical data to fellow soldiers" [1]. The article summarizes the general's bottom line by saying,

> "The network is now the Army's highest modernization priority," he says. Having every soldier plugged into the tactical network and giving them means to access and distribute information would give the Army a "tremendous advantage that we never had before," Chiarelli adds. [1]

The distinction must be drawn between having information and having situational awareness. Simply having an Internet connection does not create situational awareness. Certain features available on the Internet may be able to help, but access to a network is only an enabler of situational awareness. There is more to being aware than simply possessing a means of information availability—it is more than just raw information.

Again, to have situational awareness, the right information must be delivered to the right people, at the right time, in the right way. Good situational awareness is highly context dependent. When entering a building with Marines already inside, suddenly the context changes from needing to know information about the surrounding areas outside the building to needing to know information about the where Marines are inside the building. This type of context-based awareness is not a novel concept. Google Maps performs the same function when the detail of a map increases as the scale of the map decreases. Google decides to not burden the user with unnecessary information until they are on the scale in which they need it. This type of data-analyzed, context based awareness is programmable and can be brought to battlefield technology. Throughout this paper many more examples are used to illustrate other points, but they all share the important idea of context based situational awareness.

Like network-centric warfare, communication is inseparably connected to situational awareness. The information that creates this situational awareness is exchanged through pathways of communication. In the information age, the importance of wisely utilizing communication becomes even more important. As the ability to network becomes more prevalent, a push toward cloud-based services has grown. These cloud services gather all data on “the net.” This is neither communication nor situational awareness, this is data overload. There needs to be a focus on the distribution of the right data to the right people—data communication management. The Marine in the field has neither a network nor data management resources. That communication and management component is investigated through a network for dismounted infantry.

Figure 1 is an excerpt from a project [7] that shares the common goal of networking the battlefield. The acronyms are not important in this paper because they represent low-level physical properties of the individual networks. The project’s concept and scope are depicted by the scale of this example. With the ability to communicate data seamlessly from one platform to another helps promote the situational awareness of all of the units. It also illustrates the need for platforms dedicated to the communication infrastructure. This particular figure is from a previous program attempting to design Government Furnished Internet Protocols—a topic beyond the scope of this thesis.

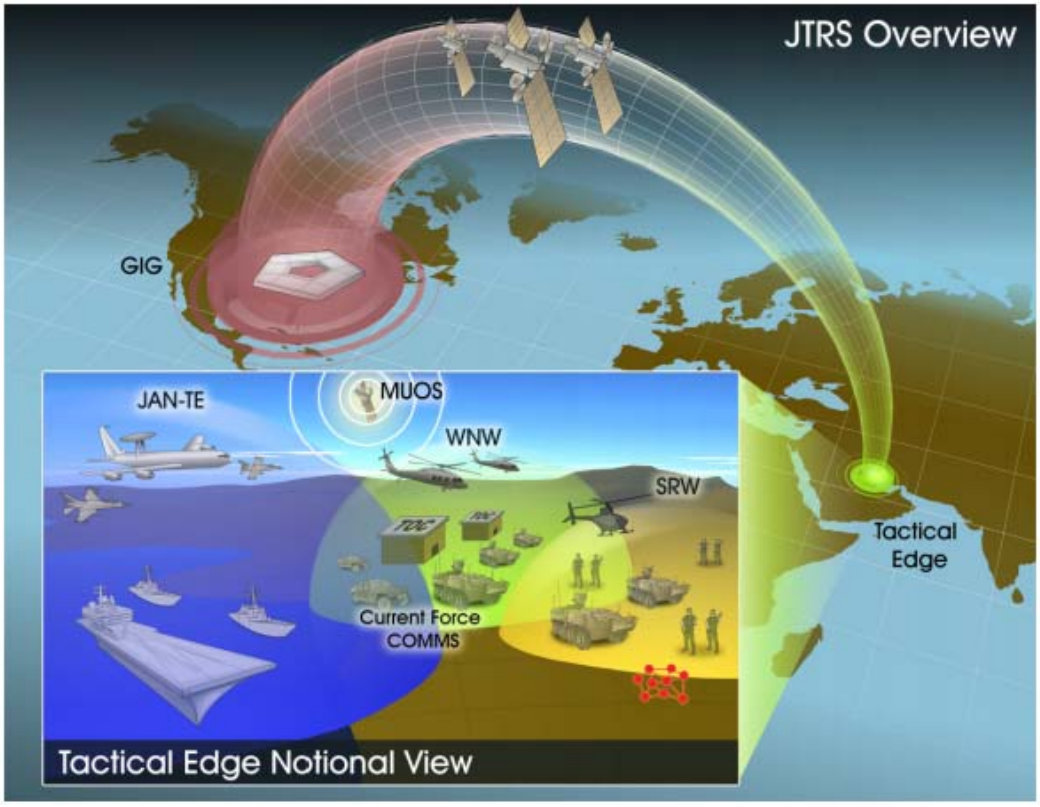


Figure 1. A pictorial representation of the scale of these theoretical networks, from [7].

Having this shared intelligence and data keeps every involved party on the same page. They all see the same information delivered in nearly the same format. There are no situations where questions over the radio verbally describe what is happening on the ground or try to verbally navigate a squad through an area. Imagine having to call Google on the phone for information. Google would have to talk through the problem and describe the information verbally. Currently, that is how those in the field are retrieving information from the larger scale networks. The final idea—the vision for these types of projects—is that these two parties, separated by any physical distance are able to have access to the same exact data and share the same situational awareness. This shared situational awareness supplies all parties involved with the information necessary for good decision making.

## C. DECISION MAKING

The mission effectiveness outlined in Tenet 4 of network-centric warfare is directly affected by the quality of decision making and control over assets and the battlespace. This last tenet enables and improves the ability of leadership to make decisions and maintain control. As a decision maker and controller, there are three questions constantly being asked on the battlefield:

- "Where am I?"
- "Where are my forces and other friendly forces?"
- "Where is the enemy and what is the best route to attack him?" [8]

A commander is faced with making decisions that could affect his life, the life of his team, and potentially many more lives. This is an immense responsibility to carry—now imagine making those decisions with partial or poor information. Military leaders are often called "risk mitigators" because of the nature of their job to make the best decision given the current information available. Ideally, the leader chooses the best plan for each scenario. Unfortunately, the reality is that the commander is often choosing the lesser of two evils. Most often, there is no clear correct choice on the battlefield. Additionally, there is no time to stop and gather more information, whatever was gathered up to that point is all there is. These high paced, intense, high-risk decisions are better handled with more information; however, as mentioned earlier, information is not everything. There comes a point where too much information causes information overload. That is why a distinction between information delivery and context based situational awareness is drawn in this work, where the intent is to deliver context dependent, relevant information in contrast to information for information's sake.

There are two sets of leaders in war, the high level leaders commanding the war and the field leaders commanding the battles. The command and control architecture, as a high level leader, plays a critical role in decision making for warfare. These decisions made by command and control cannot be made without a clear concept of what is happening on the battlefield. There are two perspectives of uncertainty in these situations. First is that Fog of War associated with the higher level leadership being unaware of what is specifically happening on the ground. Second, the decision makers on the ground

suffer from the same lack of certainty about their environment. They have an incomplete picture of what is happening in the battle space around them—information that higher level leaders may have. There is a Fog of War for both the distant command structure as well as the unit on the ground. The focus of this thesis is on improving the abilities of the small unit operation in environments with no preexisting infrastructure.

As mentioned in the previous section, situational awareness is a prerequisite to making good decisions on the battlefield and within command and control. Marine Corps doctrine expressed in the manual entitle *Warfighting* expresses the following opinion on decision making: "Decision-making is essential to the conduct of war since all actions are the result of decisions or of non-decisions" [6]. Additionally, a report published by Northup Grumman describing the use of networking technologies in combat says, "The outcomes of battles, the fates of armies, and even the survival of states have often rested on the ability to answer those questions[of decision making] quickly and correctly" [8]. A key part of effective decision-making is realizing how much decision time is available and making the most of that time. This emphasizes both the need to have the information, and have it quickly and succinctly. The Marine Corps *Warfighting* continues by saying,

> In general, whoever can make and implement decisions consistently faster gain a tremendous, often decisive advantage. Decision-making in execution thus becomes a time-competitive process, and timeliness of decisions becomes essential to generating tempo…Decision-making requires both the situational awareness to recognize the essence of a given problem and the creative ability to devise a practical solution. These abilities are the products of experience, education, and intelligence. [6]

The central purpose of command and control and all leadership positions in the military is decision making. Just as communication is essential to awareness, it is also essential to decision making. Decisions are not able to be communicated to implement control throughout the command without a communications infrastructure. More importantly, the information needed to inform the decision makers needs pathways of communication through which to be delivered. The purpose of the network built in this thesis is to further integrate communication technologies to the lowest level within the military in order to increase communication, situational awareness, and decision making abilities.

## D. SIMILAR IDEAS

With concepts of network-centric warfare, situational awareness, and decision making control in mind, there are several systems currently in place that address partial solutions to the problem. These systems are typically limited in scope and in their ability to operate across multiple platforms. As mentioned earlier, the size and weight of the following technologies typically rules them out for any practical use for the infantry; however, they show important progress towards the overall goal and reemphasize the values and concepts spoken of in the previous section.

### 1. Blue Force Tracker

Blue Force Tracker is used by both the Marine Corps and the Army. The Northup Grumman manual describes it is a system that networks together mobile platforms [8]. Typically, the platforms are vehicles due to the size and power requirements of the system. Each vehicle is equipped with Global Positioning System (GPS) transponders linked into wireless transponders. The vehicles share their locations with each other and display those locations on a laptop inside each vehicle. These laptops are terminals into the network. Written data can also be transmitted over the network through some preformatted message types. These formats reduce confusion through uniformity. Blue Force Tracker terminals can also be installed in camps so that command and control centers can monitor the data. The system currently uses a combination of line-of-sight broadcast methods along with commercial satellite connections [8]. Blue Force Tracker is such a success that it has spawned further research and effort into pushing this network capability to a lower level and, eventually, the infantry. Before Blue Force Tracker, information was passed by radioing GPS coordinates and plotting them manually on maps. Now, simply a look at the screen reveals where other units are. This was a huge leap forward, but it is still a very high level network. The scale of the equipment is currently too large for the infantry to be incorporated into the network. In order to improve this system, the networking equipment must be made more compatible with an infantryman’s equipment.

The drawback of this system is that it stops at the mounted vehicle level. It does not network with dismounted units. Handheld units do exist but only share data in one direction, from the mounted units to the handheld device. The situational awareness of the dismounted unit is more informed, but the mounted units have no information about the infantry. Essentially, the infantry can see the tanks, but the tanks cannot see the infantry. Figure 2 is a screen capture of a Blue Force Tracker terminal that demonstrates the common operating picture that the system delivers. The screen capture is an overhead image of the operating area with friendly units marked as various blue shapes on the map. The red shapes depict enemy positions that have been entered manually. The enemy's position is typically an estimate, but it helps broaden every user's situational awareness. A newly planned version of Blue Force Tracker seeks to integrate more vehicular platforms and aviation platforms, but there is currently no intention to integrate the infantry. Another disadvantage is that this system is proprietary in nature, and the data from it is confined to company made terminals that must be purchased from the approved manufacturer. Any integration to this system is limited to other systems made by the same company. There is no plug and play interoperability.

In the big picture, this system answers several situational awareness questions and allows for coordination amongst mounted units. Context dependent situational awareness is achieved through a shared common operating picture. Each mounted unit can see itself with respect to the other units on the field. Historical and area dependent information can also be accessed such as historical ambush areas or historical improvised explosive device locations. Information shared on the network goes to all terminals and messages from command and control can be distributed down to each terminal. The functionalities of Blue Force Tracker outline those abilities sought after in the smaller scale tactical network this thesis has built.

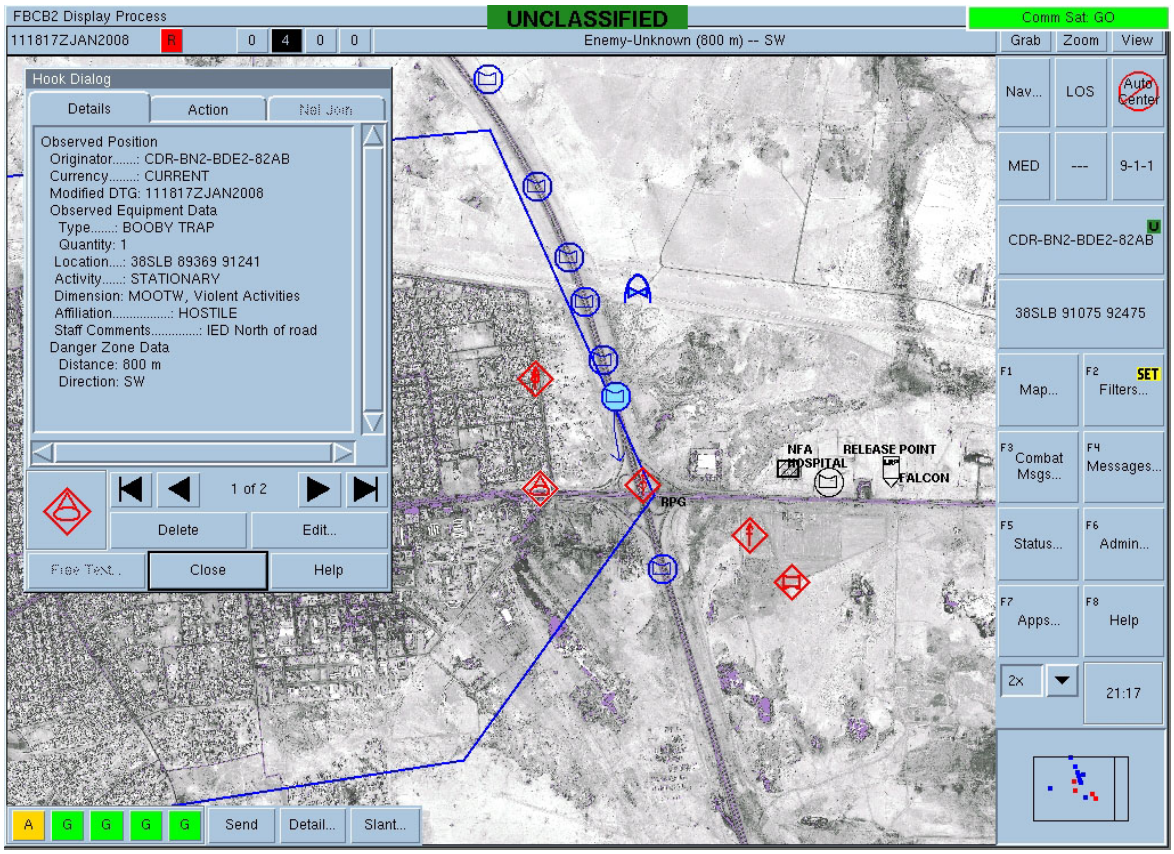


Figure 2. Example of Blue Force Tracker screen, from [9].

## 2. Link 16

Link 16 is a system of networking predominantly used by the United States Navy to share information among ships and aircraft. Much like Blue Force Tracker, this system integrates and shares data with others on the same network. It has gone through many variations and upgrades over the years. The type and format of data sent over this network is preselected and preformatted. This limits the adaptability of the system to the preselected and pre-conceived ideas of the programmers. Any on-the-fly changes or restructuring is not possible. This means that in order to create and use new message formats, they have to be upgraded within the system's operating software [10].

Link 16 is also hardware specific and uses time-division multiple access protocols as well as frequency-hopping techniques to provide protection from signal jamming [10]. This brings a very robust feature set to Link 16 which can have 127 simultaneous nets operating [10] as depicted in Figure 3. What is most interesting about these 127 nets is

that they are created separately among connected devices, allowing each network to be separate. Different operations can operate in different nets to keep data among those who need it and keep it from interfering with other missions. This concept is emphasized during discussion about the prototype design in Chapter IV—the ability of the network to deliver data to those who need it without burdening those who do not. Though many devices can be connected, all connection points are proprietary hardware for Link 16. Very limited interfacing exists for including or connecting to other networks. The common operating picture that Link 16 creates can only be shared across Link 16 approved platforms


Figure 3. Link 16's ability to have simultaneous networks, from [10].

There are two important operating characteristics of Link 16 germane to this thesis. First, Link 16 operates "without a single point of failure" [10]. It has predesignated slots in which each participant communicates, and the network continues to run even with participants missing [10]. The data sent through the network only gets to members currently on that same network. Remember that there can be 127 simultaneous

networks. The ability to isolate the data to a single operating party is important. A feature that these networks are lacking is the ability to store temporary data. There is very limited data update service for those arriving late. If the data is not broadcasted repeatedly, the late arrival or poorly connected party loses the data. Very similar to a phone line conversation, the Link 16 system is live and has limited data retention. It can also be thought of as an email account without an inbox. If the person is not signed onto the network to receive the mail when it is sent, the message does not queue.

Second, Link 16 has the ability to offer "multi-netting" [10]. This feature allows multiple participants to operate in the same network but to isolate their traffic to selected parties. This is similar to the concept of phone calls on a house line. Multiple lines can be stacked up into a single physical line, but each participant's call is isolated from the other. This is in contrast to what is commonly called a party line. The concept of "multi-netting is illustrated in Figure 4. This concept is revisited in the discussion on network infrastructure for the tactical network built in this thesis.

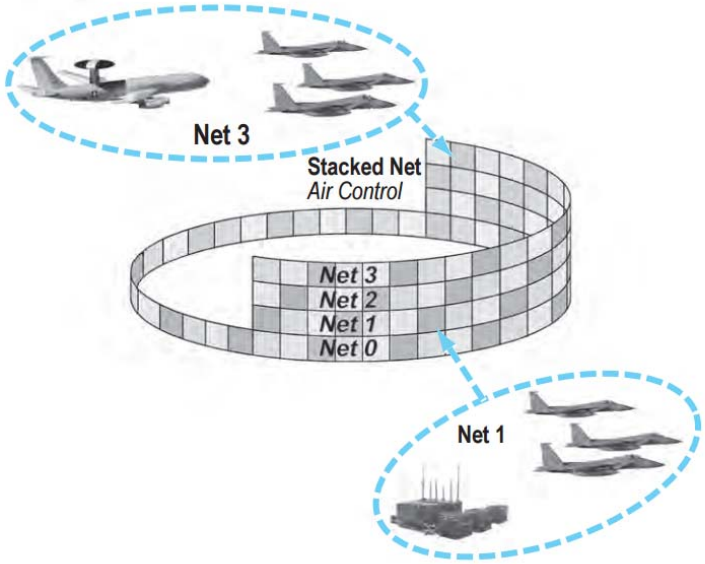


Figure 4. Link 16's "multi-netting" ability avoids the problems associated with a party line, from [10].

Link 16 has some distinct attributes that separate it from Blue Force Tracker and deliver different answers to questions related to this research. The robustness of the network is a great advantage over Blue Force Tracker, it is able to expand and have multiple people on the net at once. They can be isolated to their individual operations or shared together in an overall picture; however, the connectivity still does not connect any dismounted platforms. This information can be monitored on a station back at base and the data relayed via radio to the infantry, but this is slow and inefficient.

Several attributes of the system cause it to be ill-suited for application with infantry. First is the limited number of participants within a particular net. Due to the number of available slots, the participant numbers are most functional with less than ten individuals on a given net. This number is too small for the scale of a battalion or even a company. In addition to this is the size, weight, and power requirements of the system. They are simply on a scale too large for an infantryman to practically carry.

Even though Link 16 can be used to relay data to the field, a request and response system notoriously has problems of inefficiency and waste. Simply relaying data from these networks via a radio causes tremendous problems. The limited number of participants and issues with size, weight, and power reveal Link 16 is not currently feasible for dismounted operations. For example, a field commander needs the ability to look around him and compare what he sees with his eyes to the overhead imagery he can view on a tablet. The ability to deliver information to the lowest level in the same way it is delivered to the highest level is lacking in both Blue Force Tracker and Link 16.

### 3. Force XXI Battle Command Brigade and Below

Force XXI Battle command Brigade and Below (Force XXI) is a subsidiary product of Blue Force Tracker. It was an experiment by the Army to add more functionality and options to the system. The two systems share many similarities. In fact, it has recently been authorized to integrate these two systems. The next iteration of Blue Force Tracker carries the new name of Force XXI. It adds higher data rates, larger bandwidth, more secure encryptions, and better functionality for the user. With this new

integration, the Force XXI system includes many new interfaces with the network in addition to the typical laptop screen found inside mounted vehicles as seen in Figure 5.

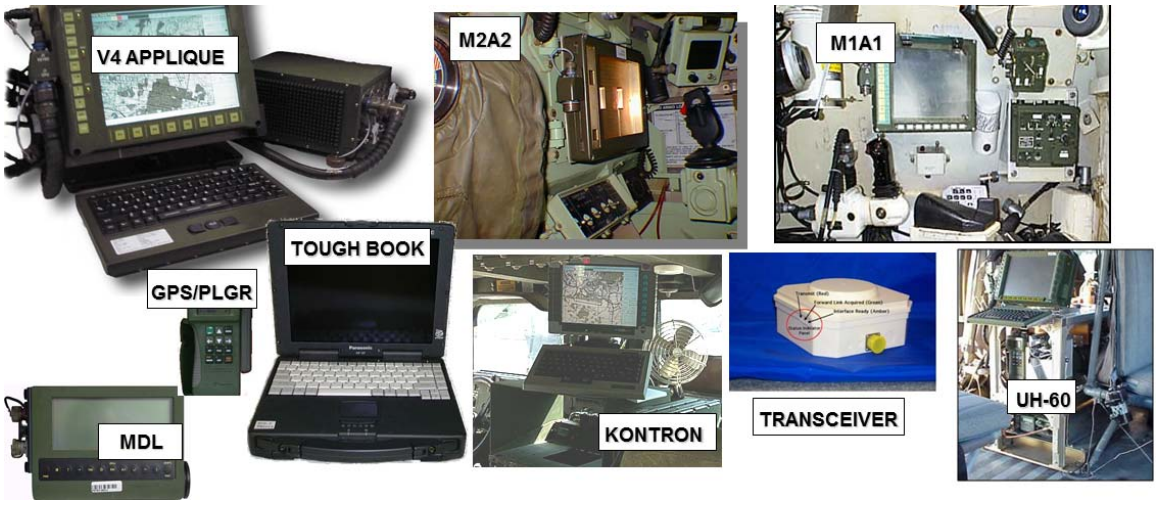


Figure 5. Example of various interfaces to the Force XXI/Blue Force Tracker network, from [11].

Because Force XXI follows in the structure and build of Blue Force Tracker, it suffers from the same shortfalls. It does not integrate lower than a vehicular unit. There is rumor that the system will share functionality with the Link 16 architecture, but it is too early to say whether that will happen. If it does, then the large-scale multi-netting ability will be brought to the battlefield, allowing for greater expansion and the ability to incorporate more devices on the same net al.l of these systems are steps in the right direction but need to be taken further. Their development is evidence of the need to develop a system that starts by integrating at the lowest level. If platforms exist that can integrate infantry units, they can certainly be implemented on vehicles later.

## E. SCOPE OF THIS THESIS

These platforms are an extension of the idea that in order for there to be organization, there must be good communication. Whether from the battlefield or a raging forest fire, the fundamentals of handling chaotic situations are the same. Information needs to be passed quickly so that the individual unit leaders can make informed and timely decisions and the higher echelon leaders can allocate and reorganize

resources. Technology has helped progress this idea, and now that the form factor and prices are dropping, it is time to integrate the lower level, smaller units to the larger force. Not only must they be connected, the data and information must be passed effectively to all those who need it.

Details relevant to networking an infantry squad are the focus of this thesis. Many of the concepts and applications of a network such as this are taken from the systems cited in the previous section. As the emphasis shifts from a large-scale network to a small-scale network, the motivation stays the same, but the problems change. The specific technical questions researchers are currently dealing within the realm of small-scale mobile networks and how that information can be applied to this thesis are outlined in Chapter III.

## F. CHAPTER SUMMARY

Today's world is a technological hive of applications, information, interfaces, and networking. Phones can be preprogrammed to send your coffee order to the barista as the phone is carried through the door. Payments can be made and tracked on the same mobile device. Tips can even be given to the barista with the tap of a cell phone screen. All this can happen while the phone is doing so much more *at the same time*. This is the context aware situational awareness that we want. Modern day technology allows the user to be alerted of a traffic accident that requires an earlier departure time to make it to the airport and make a flight. It provides the ability to renew and download audiobooks from the library, pay bills, and order groceries. It even reminds the user, based on location, to stop at the store before going home. This connected infrastructure and capability should be brought to emergency services, disaster relief, and the military.

This type of situational awareness, autonomy, automatic response, and control is essential to these types of organizations being successful and effective in what they do. Imagine the mega-online retailers trying to conduct business without the interconnections and communications they have. Those operations have the advantage of preexisting infrastructure, but when the military responds to crisis they are operating in areas with denied access to Internet, power, and cell phone towers. By their very nature, these

operating environments are more difficult and arduous than those experienced in the residential United States.

The pathway to employing communications and enabling better situational awareness is through the principles of network-centric warfare. The tenets were laid out early in this chapter, and the technological goals of the program were outlined. Why these technologies are important and also what kind of functionality they need was discussed. With the guidance of these tenets, research was guided toward most effectively delivering networking technologies to the military.

Several systems currently exist that try to provide better communications in order to help the military to accomplish its objectives quickly, safely, and efficiently. With programs like Link 16, Blue Force Tracker, Force XXI and others, there is some level of integration, networking, and data dissemination. What is yet to be done is to bring that capability to the individual warfighter—to the individual on the ground. Currently, they have radios and simple GPS coordinate readouts. All of the overhead data and intelligence is relayed to them via that radio. They are essentially operating in the field with a 1990s era pager. They can get small bits of information sent to them and can request small specific bits back, but there is nowhere near the level of integration that they could have given the right technology.

THIS PAGE INTENTIONALLY LEFT BLANK

# III. LITERATURE REVIEW

> Strategy without tactics is the slowest route to victory. Tactics without strategy is the noise before defeat.
>
> —Sun Tzu

## A. RELATED WORK

Many universities, companies, government agencies, psychologists, researchers, even hobbyists are seeking to increase the amount of integration between the user and his environment through electronics. The push to connect every device and person onto the Internet is called the Internet of Things. Today industry, research, and individuals are driving more and more toward this with smart devices from coffee pots to watches. It allows the user to be more connected to the environment around him than ever before. It is presumed that a network of sensors help increase productivity, effectiveness, and efficiency of day to day life. Specifically, this surge in research in technology has a significant impact on technological solutions to the problems discussed in Chapter II. Any avid military fiction reader or video game player has seen the glimpses of future technology. Over-head radars depict the locations of fellow units on a Heads-Up Display (HUD), and a second window on the screen depicts friendly unit's health status. All sorts of pertinent information is displayed succinctly and simply for the user to see and react to the changing environment. There are currently many existing technologies bringing that dream to a reality and even more being researched by other institutions.

### 1. DARPA Squad X Initiative

The Defense Advanced Research Projects Agency (DARPA) has recently called for research on a project they currently call the Squad X Initiative. DARPA outlines their desire to bring new mobile technology to the dismounted infantry squad. They have various specific deliverables they would like to see achieved but leave the methods for achieving them open-ended. Their request asks for several components, but one of them is the aim of this thesis research—the communications network. The Naval Postgraduate School is seeking to answer DARPA's call for research through a program called Reticle.

DARPA specifically calls for enhancement and augmentation to dismounted squad abilities, and to do this they call for research and progress in the following areas.

1. "Precision Engagement—enable the rifle squad to precisely engage threats out to 1,000 meters while maintaining compatibility with infantry weapon systems and human factors limitations."
2. "Non-Kinetic Engagement—enable the rifle squad to disrupt enemy command and control, communications, and use of unmanned assets to ranges greater than 300 meters while maneuvering at a squad-relevant operational pace."
3. "Squad Sensing—enable the rifle squad to detect line of sight and non-line of sight threats out to 1000 meters while maneuvering at a squad-relevant operational pace."
4. "Squad Autonomy—enable the rifle squad to improve their individual and collective localization accuracy to less than 6 meters in GPS-denied environments through collaboration with unmanned systems maneuvering reliably in squad formations." [12]

These are very clear and technical performance metrics. At the Naval Postgraduate School, these various goals are broken down into several different disciplines. Particularly sought after in this thesis work is an underlying technology that is not directly called for in this list but is a requisite for these technologies to be feasible. The focus of this thesis is to create a network that can operate independent of infrastructure and provide the communication capabilities required to achieve synchronization for the capabilities listed above.

### 2. MIT Response

Several universities and research centers have responded to this call for research. One example is the Lincoln Lab at Massachusetts Institute of Technology (MIT). This lab has several ideas for Squad X that are similar to ideas laid out in this thesis. Lincoln Lab's desire is to deliver on all of DARPA's requests. They seek to deliver a series of products to meet the design requirements, and they have various ideas for solutions to the design challenge including networks, sensors, and components designed to integrate with existing platforms.

A concept of MIT's overarching design and how they seek to enhance the situational awareness of the units on the ground is shown in Figure 6.

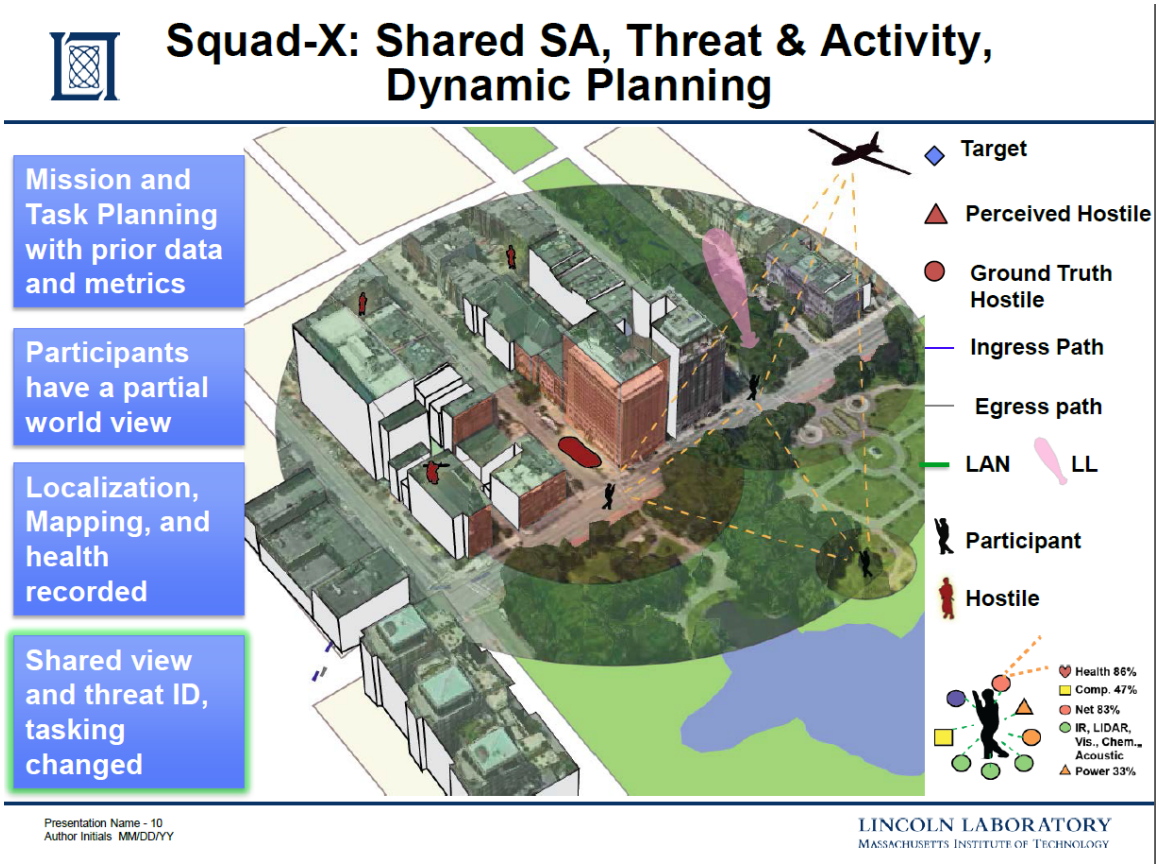


Figure 6. Lincoln Labs concept of the Squad X final product, from [13].

This is accomplished through several methods including map integration, shared health status of members, shared locations of friendly and enemy units, and more. In order to achieve these abilities, several factors must be considered, including the backbone of the system. That backbone is the devices carried by the individuals that run both the network and the various sensors and equipment also carried by the individual. An example of the various pieces of equipment that need to be carried by the individual in order for this system to work is shown in Figure 7.

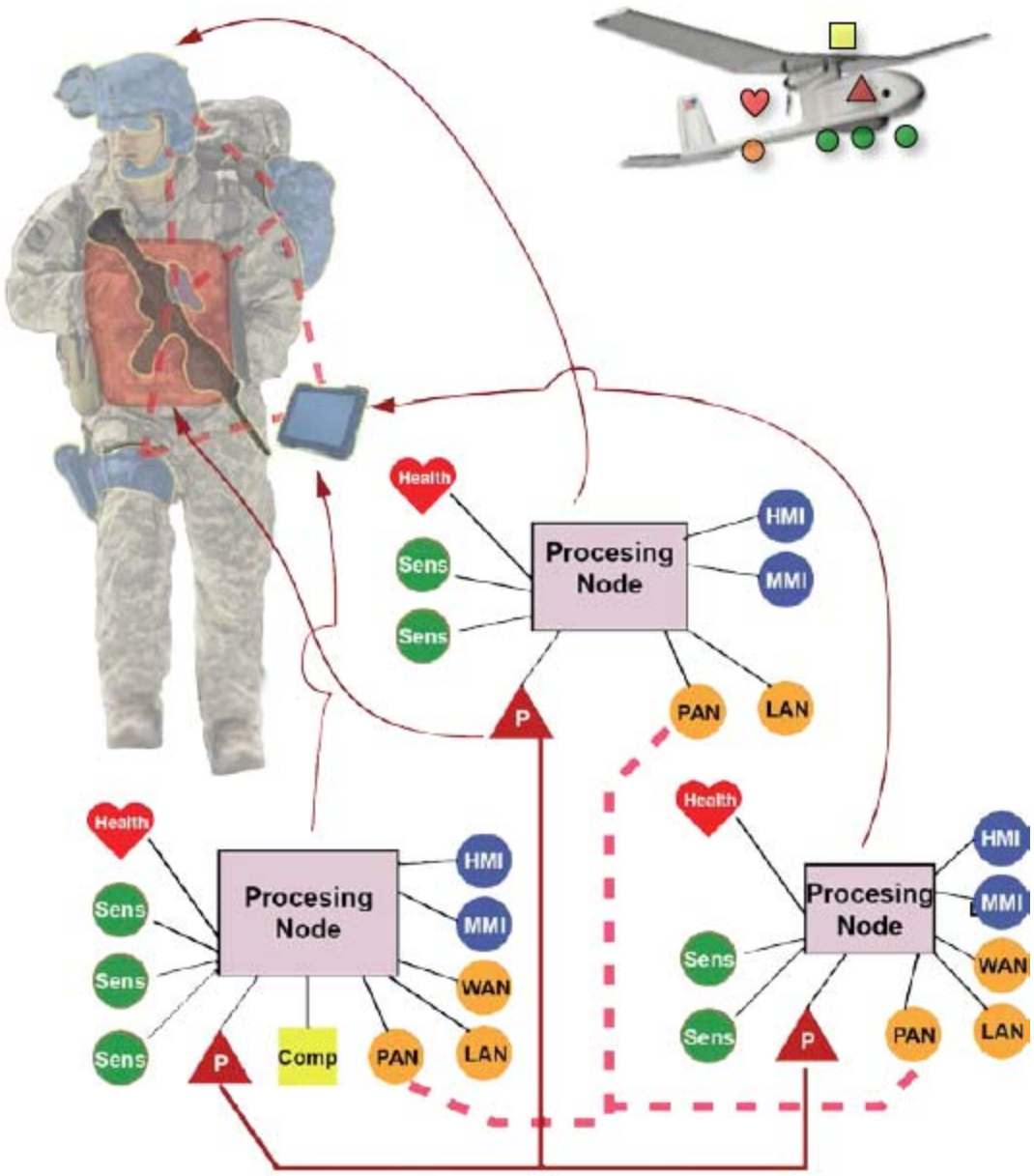


Figure 7. Example of soldier personal sensor and networking capabilities, from [13].

The specifics of what is depicted in Figure 7 are not as important as the message of complexity that it conveys. The message this delivers is that before there can be a discussion about all the capabilities the system will have as a whole, there must to be a discussion about the components that will run the system.

The ideas from both the DARPA initiative and MIT's research are great examples of the abilities and applications that will be built to further develop the communications network developed in this thesis. The key to enabling these types of features and abilities is through that network. Without the ability to share information, this shared, contextual situational awareness is not possible.

### 3. Naval Postgraduate School Response

There are several theses and projects that are currently seeking to answer DARPA's call for research at the Naval Postgraduate School; they are organized under a group called Reticle that seeks to deliver technological enhancements to military personnel. This is one of the group's first projects.

#### *a. Inertial Navigation Systems*

The fourth goal DARPA lays out deals with determining the physical location of individuals within the squad. Global Positioning System (GPS) is the typical solution for location technology, but there are times when GPS does not work or is unavailable. Navigation technology that can operate in a GPS denied area has been sought after for a long time. Inside buildings, caves, underwater, and some parts of the globe are all unreachable by the GPS satellite signals. A new mobile, networked squad needs to be able to operate in the absence of GPS. It can have the option to use GPS but cannot be reliant on it.

Inertial navigation is a possible solution for operations in GPS denied environments. Inertial navigation uses the concept of tracking the motion of the object and being able to tell how displaced it is from its original starting point. It uses that initial point as the reference frame. That initial point can be manually inserted or synchronized off of GPS. The challenge of this technology is to accurately measure and record the movement of the body. Say, for example, the device starts in a doorway and is then is carried two steps forward and then three to the left. Inertial navigation says that the device is at coordinates (2,3). Comparing those coordinates to a map shows that the device is in the kitchen. Just as GPS shows the coordinate positions, so does inertial navigation. Because of the fact that inertial navigation must measure the physical position

by measuring the movement of the body, it is very difficult to isolate and read these signals. Each small offset from what is actually happening causes compounding error to propagate through the system. This error is often called drift.

Many technologies and platforms in the military already use this technology. It is very sensitive and cumbersome. It certainly is not mobile and portable technology that an individual can carry comfortably in the field. It is currently used on larger platforms to accurately estimate position based on movements. The system shown in Figure 8 is a Raytheon Marine Inertial Navigation System (MINS). The main part of the system is roughly nine feet by nine feet, with two other components of slightly smaller size. These systems lend to the credibility that an inertial navigation system does work and is feasible.


Figure 8. Raytheon's Marine Inertial Navigation System, from [14].

Researchers at the Karlsruhe Institute of Technology (KIT) and the Institute for Information Processing Technologies (IIPT) are producing solutions for personal navigation that share the same principle as the larger systems. They use a sensor to measure the amount of movement and then track that movement over time. Most research uses accelerometers and gyroscopes to estimate that movement. This becomes more difficult when building a system that works for every individual. Because of the uniqueness of each person's walk, it is very difficult to program a system that works for everyone. In addition to that, the software must be able to handle terrain changes and irregular speeds without causing more error. So far, many of these researchers have had very good results and there are even some forms of products available commercially.

An example from the KIT and IIPT experiments is shown in Figure 9 [15] that shows the size of these sensors in comparison to a coin roughly the size of an American penny. Some of the results this research group has achieved in the accuracies of their navigation are shown in Figure 10 [15]. KIT's research reflects the progress that most research groups are achieving in an infrastructure independent indoor navigation solution.

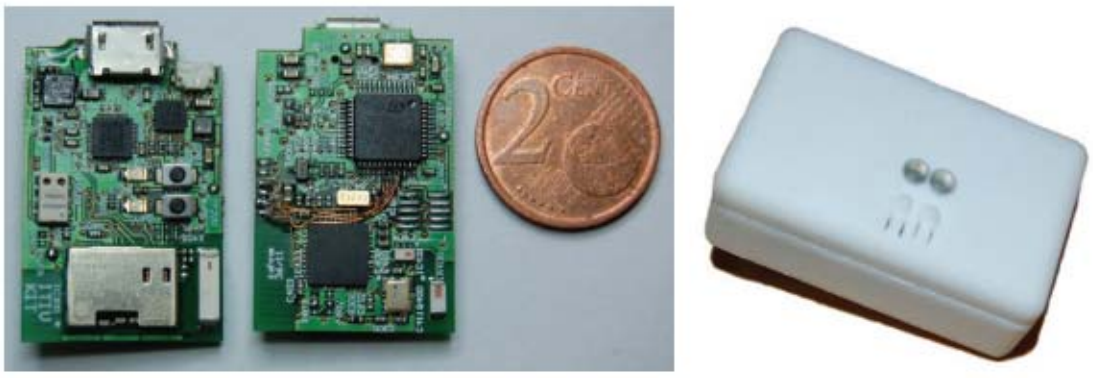

Figure 9. Example inertial sensor size, from [15].

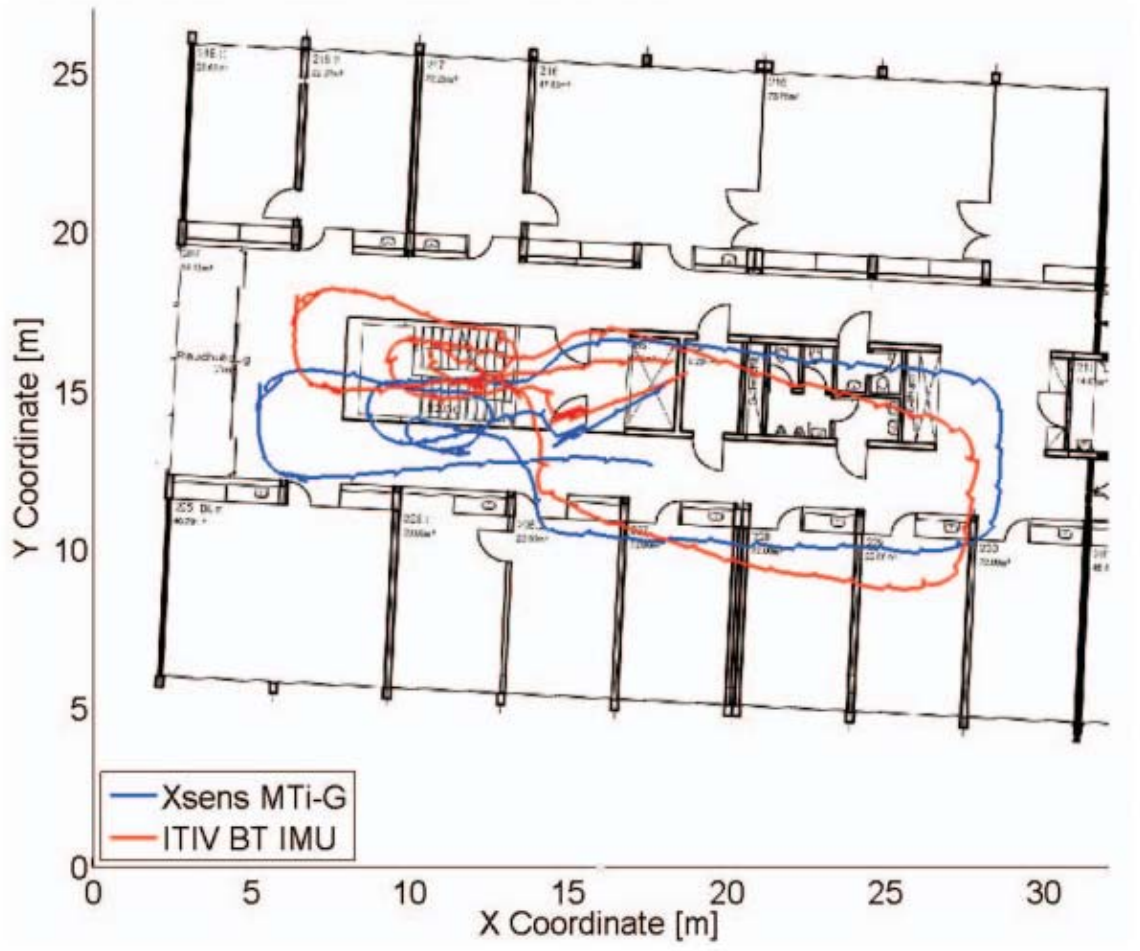


Figure 10. Example of localization accuracies achieved by KIT, from [15].

These researchers have had measurable success with achieving accurate indoor localization. Much of the difficulties currently arise from the inaccuracies of the physical components. As the sensors become more accurate and capable, the software analyzing the movements becomes more accurate. This specific research reflects the progress of most groups in these areas. It is a difficult problem that is likely to be solved in the near future. Additionally, thesis research attempting to solve this problem at the Naval Postgraduate School is making similar progress using foot mounted sensors to create an Inertial Navigation System.

### *b. Orientation and Position Resolution*

A conglomerate of the goals laid out by DARPA is the ability to resolve the orientation of an individual in the squad electronically. This can lead to better engagement with the enemy outlined in the first goal, better sensing in unseen areas as outlined in the third goal, and also informs various other features of the network. Similar to the problem of knowing an individual's position is that of knowing his orientation. Orientation contributes greatly to the overall picture. Just as Google maps can predict the direction users are traveling on the highway and only show information in front of the user, the same holds for the value of knowing the orientation of an individual. When looking a certain direction, information about that direction should be posted to the output. Any information about other objects around the user is superfluous and often unnecessary. Sometimes, this question of orientation can easily be determined by the direction the object is moving, but this is not always the case. There are many military circumstances in which the direction the rifle is pointing is not always the direction the individual is moving and vice versa. There are also chances of targets remaining stationary for periods of time.

The Swedish Defense Research Agency has been trying to solve this problem by means of a small-scale GPS-like system. Their concept is depicted in Figure 11.

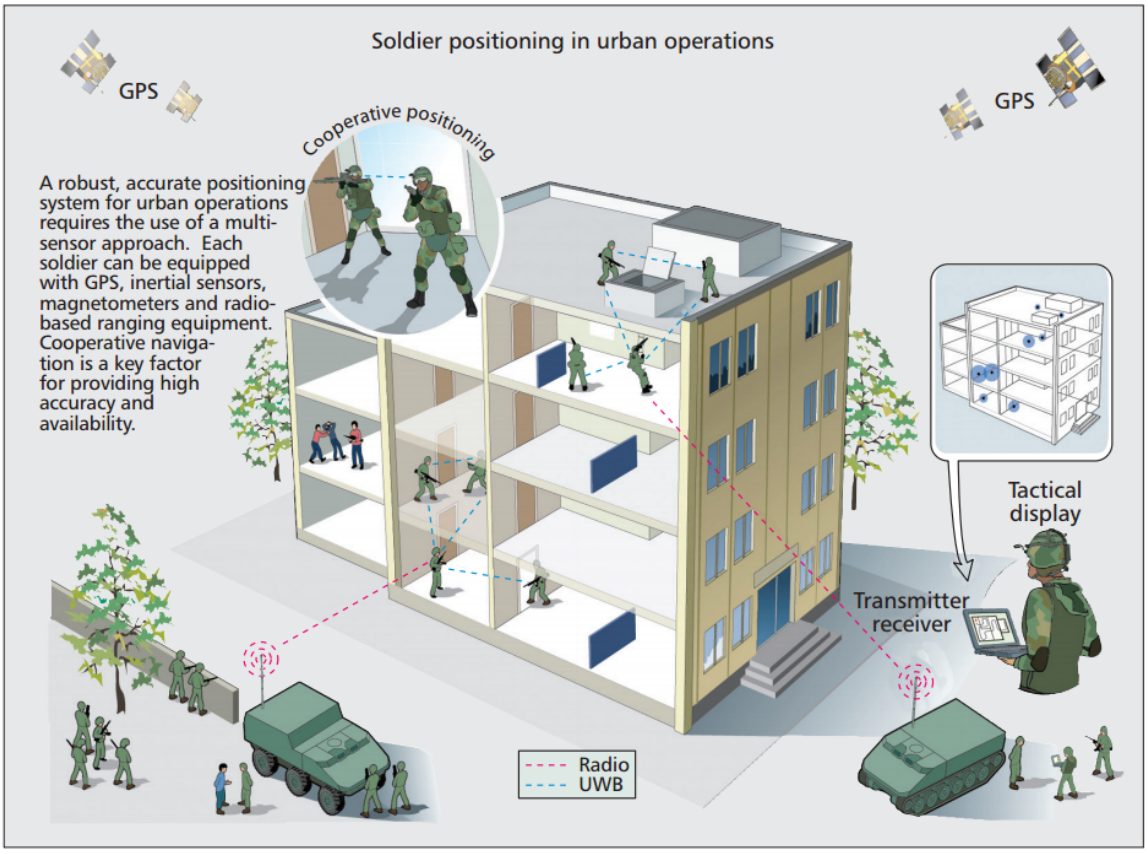


Figure 11. Example of research finding pose and position of individuals, form [16].

In having exterior platforms that are in fixed positions, like GPS satellites, they can communicate with the units inside the building to determine location in three dimensions [16]. They combine this exterior GPS setup with inertial navigation sensors to determine the exact position and direction of the units.

There is also an ongoing project at the Naval Postgraduate School to determine the pose of a rifle via an inertial navigation sensor package and avoid conflicting fields of fire. Discussed in Chapter IV, this project is later integrated into this thesis for a test of both the connectivity of the network and the reliably of the project's mathematical transforms.

## B. POSSIBLE NETWORK ARCHITECTURES

With an overview of the features that a forward operating squad may need, the next question—the question DARPA is asking—is how this network works. First, some knowledge of the network topologies and abilities are covered. Then projects attempting to employ these technologies to the problem are reviewed. Their shortcomings are considered followed by plans for this thesis research to overcome those gaps.

### 1. Mobile Ad-Hoc Networks

Figure 12 is an illustration showing that the MANET comes in many shapes and sizes, from hundreds of nodes to individual networks between a few persons. All of them have the common goal of distributing information among its nodes discretely, securely, and effectively. Because of this, they all have different wavelengths, bandwidths, and protocols. The push for more MANET technology in the military has come predominantly from the doctrine of network-centric warfare. For this thesis, the concept of small-scale, squad-size groups of individuals connected by a MANET is key. Throughout the rest of this thesis, attention is drawn to how this need for small-scale MANETs was satisfied in this work.

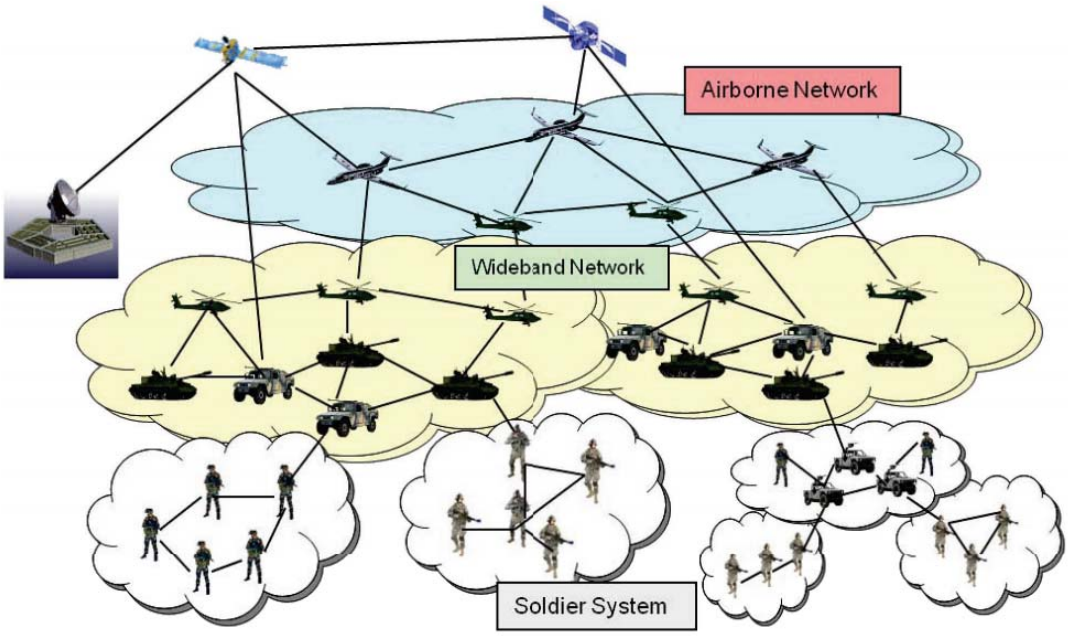


Figure 12. Example of different sizes and scopes of MANETs, from [17].

The MANET can be thought of as a combination of two things. Typically, but not always, it uses the IEEE 802.11 Wi-Fi standard of communication but instead of having a central connection point such as a Wi-Fi router, the nodes are connected directly to each other. Ad-hoc networks are typically used for special purposes, such as, transferring files from one laptop to another, connecting to a printer to quickly print documents, or connecting a phone to a desktop to download photos. These networks are formed for a single purpose and usually without external infrastructure. Adding the characteristic of mobility to the ad-hoc network creates a MANET.

## 2. Publish-Subscribe Network Schemes

Publish-Subscribe architecture is very different from traditional client-server communications found in most IP based Wi-Fi networks. This new architecture is controlled by a component known as Data Broker Messaging or just the Broker. In many ways, the Broker can be thought of as a smart router. In a typical IP network, the message comes into the router and the destination address are stripped off the front of the message. The addresses are looked up in a table that lists where the addresses are physically located before they are sent off. The messages pass through the router, much like letters

through the postal service. The contents of the messages are not read nor is there any information on the data contained within the message. The router simply forwards the messages based on destination addresses.

In a typical IP routing scheme, all traffic is sent with its final destination in mind. Every IP packet coming into the router *must* have a destination address and a source address attached to it, just like an envelope at the post office. A Data Broker scheme is different because it acquires and categorizes the data from data producers called publishers. The Broker then redistributes data based on demand to the data consumers called subscribers. Data is given to those who request or subscribe to it rather than those to whom it is addressed. It provides a mechanism for free-flowing information being instantly available to all interested parties without the need to rebroadcast to every user [18]. Messages are sent based on their content in contrast to being sent to a specific user. That way the message is published and available to anyone that is interested in the subject. The sender does not need to know that the recipient exists. The distribution is handled by the data broker instead of the data originator. This means that the sender and the recipient are anonymous to each other.

A second difference is in a protocol that is called Time-To-Live (TTL). It is a tool used in IP routing to prevent a message from infinitely traveling through the network. A router knows approximately how many device hops a message needs to take to get to its destination and attaches a TTL to the message so that in the event that it does not make it to its destination it self-deletes. This may happen if the end device or final destination is not currently connected to the network or is unreachable. The Data Broker does not have this problem because it is able to hold the data and disseminate it at any point. The data has no expiration date on it [18]. Again, this allows the senders and receivers to be independent of each other.

The third difference is that the data passing through the Broker is retained in the broker and can be used later in the event of a node or endpoint failure [18]. This is very similar to the argument against end-to-end communications in the network. Since the data is stored in the data broker, it is not only available to more end point users but also maintained independent of the activity of the publishing nodes. This allows for nodes to

transition from online to offline and back again without affecting the integrity of the network. Those intermittent nodes also maintain up to date data because they simply retrieve or place the newest information with the Data Broker every time they come onto the network. A single node can come on to the network and publish data and then go back offline. Separately, another node can come onto the network later and retrieve that previously published data [19]. This setup allows the system and network to work without the need for every node to be online at the same time. Imagine if emails could only be delivered if the account was actively signed into the network and that any mail sent when the user was offline was never delivered. The Data Broker method prevents that from happening. It allows for flexibility, scalability, and durability to the network. A positive feature of this is that even a partial build of the network works—the system is not dependent on having every node online at the same time. The organization ZeroMQ is a dominant leader in research and implementation of Data Broker Messaging systems. Their introductory paper on this architecture describes it as follows:

> With this style of communication, the flow of information is interest-driven rather than destination-driven. Senders simply send (or "publish") information in the network without addressing it to any specific destination. Receivers specify the kind of information they are interested in (or "subscribe" to this kind of information), without regard to any specific source. [20]

From their research and explanations, the ability of single source data to be delivered from a publisher to multiple recipients or subscribers is shown in Figure 13. The threads connecting the publisher and subscriber are setup up by the Data Broker. This also shows the nesting ability of the scheme and displays the separation of data producers and consumers. The scheme has the ability to have layers within the network that are physically unaware of each other. The data flow in this example is in one direction—from the top down; however, this system can be set up for bidirectional communication to allow for a more lateral distribution of data. It does not need to be hierarchical.

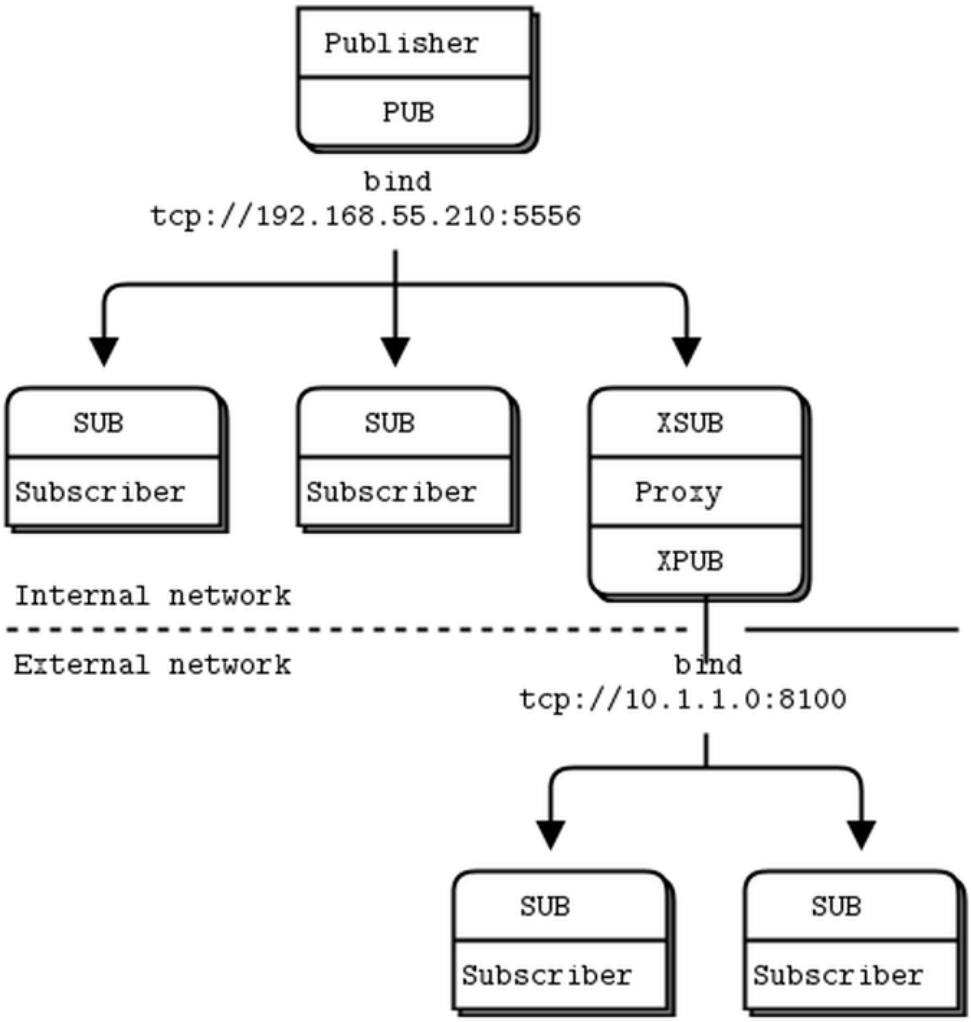


Figure 13. Example of a nested publish-subscribe architecture, from [18].

Some of these advantages can be disadvantages based on the use and setup of the system. With a managed data broker arrangement, there is a single-point of failure in the Data Broker. This potential for a single-point failure also exists in the router of an IP network. The same disadvantage of bottlenecking through a Data Broker occurs through a router; however, the overhead with a Data Broker is usually more than that of a router, due to categorizing and storing the messages. In the end, the choice is determined by the intended purpose for the network. The data dissemination achieved by a Data Broker may be a better choice for some networks but not others. If the only requirements for the network are one-to-one message deliveries, then perhaps an IP message service may be a better configuration.

The work from ZeroMQ shown in Figure 14 depicts the Data Broker as a directory service and sets up the connection lines between applications or nodes. Those lines include Q1 through Q3 that represent where the data is stored. The queues shown can be physically located where the broker is or on a separate platform as designated by the Broker [18].

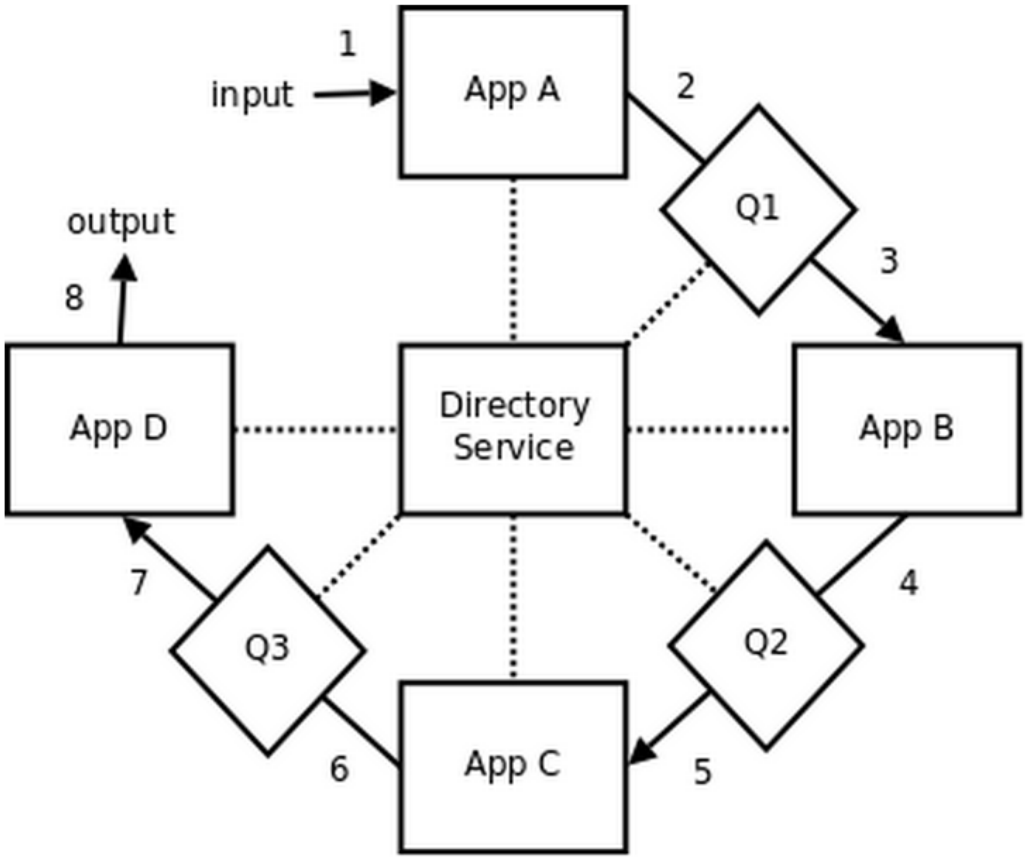


Figure 14. Example of Data Broker with message "queues," from [18]

This particular setup shows a very linear system where data flow is in one direction, but that is not always the case. Later examples in this thesis show alternate setups of this network. Data flows from App A to Q1 and then to App B, but in reality any number of applications could publish to or subscribe from any of the queues. The Broker topology itself can also be implemented in a disseminated way if the network calls for it.

Brokers can be disseminated and coexist on separate networks. These separately managed networks can still operate together, as shown in Figure 15.

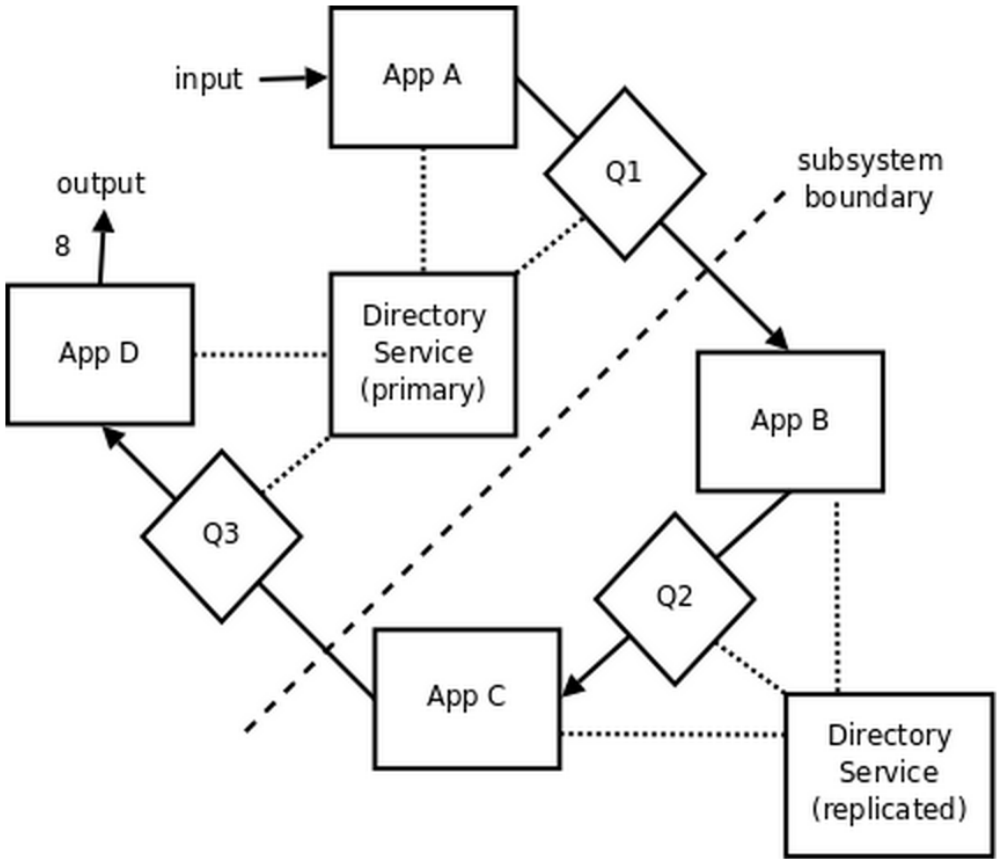


Figure 15. Version of Data Broker with multiple Brokers, from [18].

This ability to nest networks within each other allows nodes to be connected to more than a single broker but allows the network to operate through several separate directing entities. This ability is ideal for nesting a network within or below another network, fits perfectly into the model for military related operations, and allows for the ability to keep local information local while still being able to forward up relevant data.

## 3. Projects Utilizing Brokers and MANETs

Several research groups have been trying to overcome the difficulties of integrating a broker scheme with a MANET infrastructure. These difficulties arise because of the dynamic and mobile nature of a MANET. Often there are intermittent connections, distributed architectures, and varying physical setups. Two dominant theories have surfaced as solutions implementing a Data Broker scheme into a MANET architecture. First is to have every node act not only as a publisher or subscriber but also as a message forwarder to the broker. Where before each publish and subscribe node had a direct line to the Broker, now a node can send messages to the Broker through other nodes. This embodies the basic concepts of mesh networking and a unicast scheme. It adds a routing functionality to the network in the software layer and adds overhead to the network that slows the network.

The second theory is to have every node be a broker or have the ability to act as or step in for the Broker. This establishes redundancy in the system but also causes overhead that slows the network. The Motilal Nehru National Institute of Technology in India in their publication, "A Reconfigurable Distributed Broker Infrastructure for Publish Subscribe Based MANET," plans to use a process somewhat common to mesh networking and elect a broker for a given cluster of nodes. Each broker now speaks to all nodes within a single hop to itself but speak to other Brokers through two or more hops [21]. Most of the following ideas are variations of the two dominant theories.

A very novel idea from the University of South Brittany, France is to use the data brokers and nodes as opportunistic message carriers. Their publication, "Towards a Usenet-like Discussion System for Users of Disconnected MANETs," suggests that a couple nodes in a MANET would either be the data brokers or have copies of the data broker information in the event they are elected Broker. As a mobile node of the network leaves one cluster of nodes and enters another, it brings information from the previous network into that new network. The node carries copies of all the data from its old network. The transient node has now brought information from the old network into the new network [20]. They call this method, "Opportunistic, delay-tolerant networking,"

focused on "supporting content based communication in a disconnected MANET…dedicated to information sharing" [20].

When a node enters a new network, it broadcasts to its new neighbors the topics it is interested in and advertises information about the data it is currently carrying. This method is shown in Figure 16. The nodes respond to each other by distributing data amongst themselves until the data is disseminated.

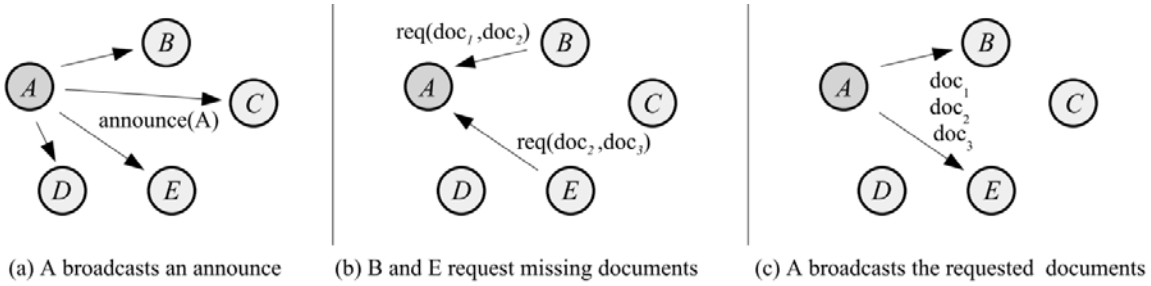


Figure 16. How data is distributed when a node leaves one network and joins another, from [20].

The data is sent through the network "on the transmission facilities of a standard IP protocol stack" [20]. This means that the data dissemination with this project is managed at the application level and is sent using IP data transmission standards. This is simply the mode of transportation in this implementation. Just as the post office has the option to deliver with trucks, carts, or by foot, this transportation method is a choice of the implementation. The data broker is still controlling distribution of the data and is still retrieving all the messages. The data broker in this implementation manages the flow of data but actually sends the data by means of the IP transmission standard. With this implementation comes interoperability and the ease of use with a familiar protocol like the IEEE 802.3 IP standard and the 802.11 wireless IP standard. This shows that the Data Broker scheme can be implemented using the physical transportation mechanism of the IP standards. That is to say, the Data Broker can be implemented on an IP network; however, the uniqueness is that even though it can be used—it is not reliant on it. The scheme can operate with an IP network and with other network implementations concurrently.

The University of South Brittany has taken what is commonly considered as a disadvantage of a MANET and used it as an advantage in their implementation:

> In a disconnected MANET, no device can be considered as being stable enough, accessible enough, and resourceful enough to play the role of a dedicated [data] server….No device in a disconnected MANET can serve as a central repository for maintaining the list of all [topics]. [20]

This opportunistic message carrier idea is explained very well in an example they have of sharing news articles among electronic newsreaders. They describe each device or e-reader as storing certain articles of news. As each device comes into contact with either a small cluster in the form of a MANET or a single other device, they advertise to each other what articles they have and what articles they are interested in. If a shared interest exists, they exchange the desired information. The advertising is done through keywords that describe subjects of interest and if there is data duplication within a device, it is simply deleted [20]. For this data dissemination process to work there are four main components on each node: application, middleware, storage, and network interface. The application in this example is the newsreader or application that displays the files. The storage device is any physical storage the individual node has, and the network interface is the hardware to interface with the network depending on the protocol (physical, wireless, etc.). Most unique is the middleware—what keeps track of the newsreader's interests and the database of news. Additionally, the middleware is aware of what articles it is carrying in storage, what topics it is interested in, and manages how to catalog and broadcast that data. The middleware acts as a common translator to categorize and help organize the data for easy advertisement and transportation. Each node must have this middleware to interpret the short hand spoken amongst them. This is the key to sharing the data within a Data Broker scheme like this. They must all speak the same foundational language. That is not to say that they need to be coded in the same language but that they must have the same formatting so that a message sent from one can be deciphered and understood by any other. In these cases, the news interests are defined by the user. The breakdown of what each physical component of the network requires in order to operate the system as a whole is shown in Figure 17.

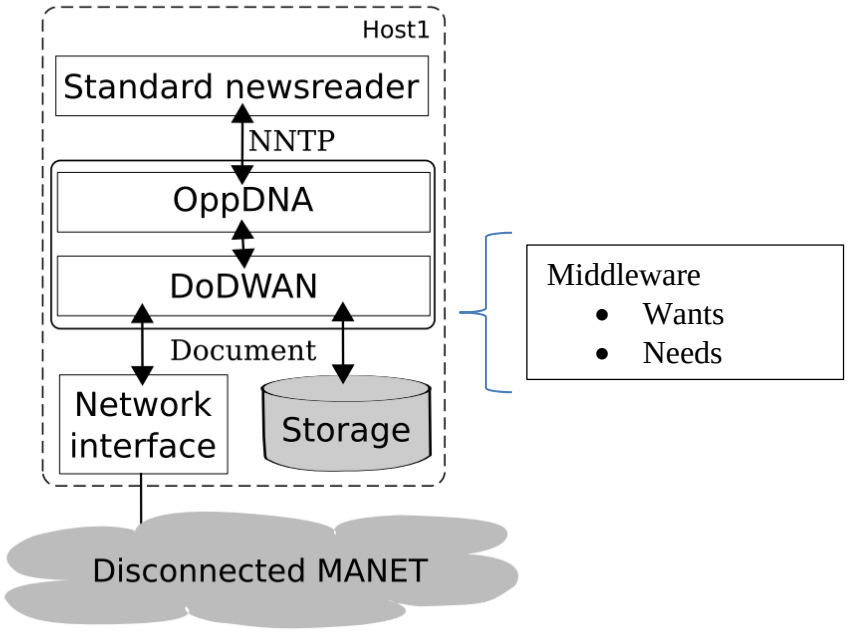


Figure 17. Diagram of middleware (DNA and WAN) managing the data for this node, after [20].

What this structure does not have is any point-to-point communications assurance. There is no guarantee that data is completely distributed throughout the network. Mission critical data cannot rely on physically moving nodes to come within proximity of each other for data to disperse. Some of these carriers must be able to send messages to other parts of the group and not wait for it to physically be carried to them. This current model is much like diffusion in a liquid. It allows the swirling of the nodes and time to completely disseminate the data. Additionally, data is only carried through the network by those nodes interested in that data. It is solely opportunistically transported. If there is no interest in that particular information, it propagates slowly through the network. A tactical version of this network needs an agitator to more aggressively mix and disseminate the information amongst the nodes so that whenever the information is needed, it is accessible—perhaps by a drone. In a small group that is not physically dispersed, this network implementation may prove very useful.

If every node were programmed to share all information, data dissemination schemes like this have the potential of becoming overloaded if the network traffic becomes very dense. There is a particular research group at Beihang University in Beijing [22] that is striving to develop a technique to alleviate that pressure by making the individual nodes broadcast based on context and situational awareness. This takes the previous model of content-driven dissemination and changes it to content-and-context-driven dissemination. “Through learning context information about the users, e.g., their interests; their mobile habits, nodes could guide their behaviors self-adaptably” [22]. They take into consideration the physical deployment of these networks. One of their first goals is to make the broadcast frequency or data refresh rate more efficient. Their idea is to have the device aware of the density of the nodes around it and adjust its broadcast rate based on that. The more nodes around it that have relevant and changing information, the faster it updates. Conversely, if a node has no neighbors, it significantly reduces its rate of broadcast. There is a twofold benefit to this, it helps tailor network traffic to fit the needs of the network and saves on power for the device. If there are no nodes to receive broadcasted information or the information is not changing quickly, there is no need to broadcast. Each broadcast consumes the finite resources of both power and bandwidth, and each saved broadcast saves those resources [22].

An example of a physical layout that causes nodes to adjust their broadcast rate based on node density is shown in Figure 18. What this does not show is a broadcast rate determination based on data change rates. An example is the contrast between a video feed and temperature readings. To provide streaming video, the device broadcasts each frame at roughly a 30 Hz, but temperature does not change very frequently. Because of this, there is no need to have the same refresh rate for temperature.

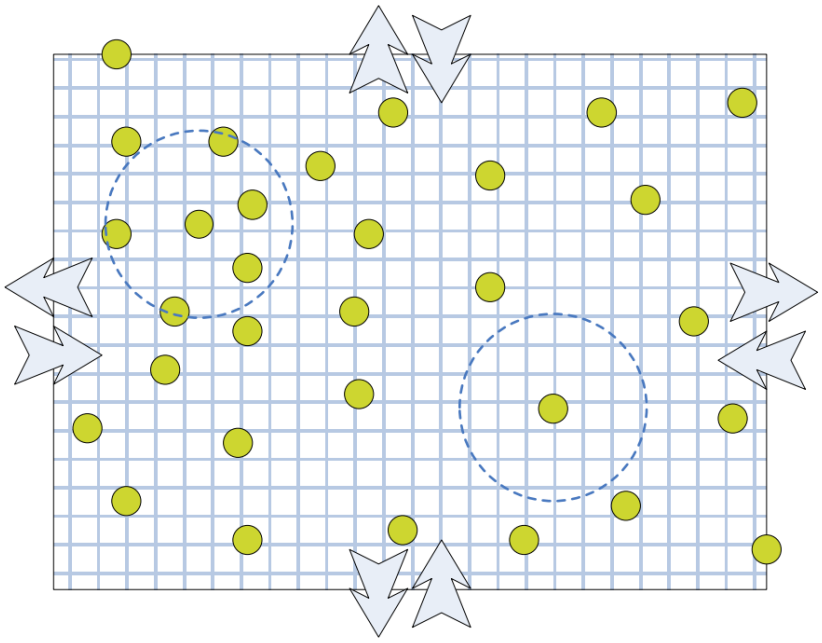

Figure 18. Example of how densities can change broadcast frequencies, from [22].

These types of concerns are important to consider when viewing the operating environments of the network. Having a fixed infrastructure inside a building is obviously the easiest to implement because nothing moves and the infrastructure can be pre-established, but the nature of a tactical MANET makes it much more difficult to plan for every scenario. Examples like this reveal the concerns that are relevant to the discussion that are not related to the actual routing or information, like power. This keeps in mind that a tactical MANET is more than just a networking problem.

## C. CHAPTER SUMMARY

The defined goal of network-centric warfare has framed the goals and brought metrics to the equation for how these technologies will work. Researchers, universities, and companies are discovering and producing incredible technology that will bring this functionality into environments that do not have the preexisting infrastructure to enable it. Many of these technologies and capabilities already exist, but the problem is that they are not currently in the proper form factor, cost, weight, and reliability necessary for forward operations.

The need is clear for a networked infantry squad, and throughout this chapter specific projects were identified and referenced in order to substantiate the need for these systems. Many of the projects point to network topologies that can implemented into a mobile network such as the network described in this thesis. Those topologies, however, have only been simulated. The goal of this thesis is to physically build a network that delivers the capabilities necessary for distributed data dissemination. Attempting to provide the ability of data dissemination among a small group of people, with the eventual integration into a larger network, is a further goal.

A test was identified in this chapter was applied to the network in order to verify the abilities of the network. That test is a friendly-fire detection program that needs to be able to share data among the distributed components. The network built for this thesis delivers that information. The goal of this thesis was to build only the network and then to test it in conjunction with other projects, not develop a fratricide detection program.

The goal was to produce a small network capable of transmitting data among members of the network. This data is transmitted in a low overhead, low power, highly reliable way. A unique solution to small network issues as well as big network data identification issues was attempted. The network was developed in such a way that a common operating picture is easily reachable by every part of the network. This is accomplished in such a way that upgrading, expanding, and improving the network is easier and more affordable than for current systems. Multiple platforms and propagation techniques are used in order to be more integrate-able with current systems. Most importantly, data is transmitted in a unique way such that data moves from the data produces to those who need it—the data consumers.

# IV. EXPERIMENTAL DESIGN

> In general, whoever can make and implement decisions consistently faster gains a tremendous, often decisive advantage. Decision-making in execution thus becomes a time-competitive process, and timeliness of decisions becomes essential to generating tempo…Decision-making requires both the situational awareness to recognize the essence of a given problem and the creative ability to devise a practical solution. [6]
>
> —*Warfighting*

Explanations are given to justify decisions and to draw connections between the components utilized and how they relate to the thesis's research goals. Each should showcase the versatility of the network to utilize and universalize the inputs of several devices so that the data is readable across all devices. This common language throughout the network enables each node to interpret data and provide situational awareness. Assumptions are emphasized and weaknesses discovered are openly discussed. It is the intention of this chapter to focus on the work of this thesis—the network. Though the fratricide detection program is used as an example of the networks ability to distribute information, it is not the focus of this project.

## A. GOAL

The goal of this thesis is to create a small mobile network for dismounted Marine infantry. As a network there are several operating requirements necessary to create the final product. First, the network needs to enable lateral data distribution amongst physically dispersed nodes. Second, there is a need to keep track of the data and have the ability to scale in both the amount of data it can carry as well as the number of nodes that can connect to it. Third, it has to be built of small, affordable, low power consuming components that do not significantly change the Marine's current equipment list. Lastly, and likely most important, is the need to effectively employ a network architecture in order to retrieve information from the data producers and deliver it to the data consumers. A unique version of a data broker is used to achieve this goal. Though this current prototype deals with small isolated groups communicating laterally among themselves, the build must also keep in mind the eventual integration of this smaller network into the

larger networks such as Link 16 and Blue Force Tracker. To enable the future integration to these larger networks, this prototype must take into account both the software and hardware requirements of these other networks. Ideally, a feature of this network is that it is a communications network that is independent of the physical link layer. This means that it can run over any medium of transport chosen by the user. It cannot run on a proprietary signal or be forced to work with a specific broadcast method. Examples of how this was done are shown. This ability provides interoperability and provides the troops on the ground with the best communication and data available.

## B. PLAN

The research discussed in Chapter III has put forth many relevant and useful ideas, but none of them have been built and tested. All of the research cited previously has been simulation. Despite this, they have in mind the same theory as this thesis—start small. After solving the small problem, the larger problem is easier to solve. It is a bottom-up approach. Specifically, the purpose of this thesis is to bring the vehicle interconnects that exist in systems such as Blue Force Tracker, Force XXI, and Link 16 down to the infantry Marine. These products now perform amazingly for vehicle coordination in large movements of armored groups. They can even coordinate general areas of friendly and enemy ground troops, but there is no connection of an individual solider into the system. The goals are first, to bring a light-weight, affordable, low-power-consumption network that can be used by squads, possibly up to battalions. The second goal is to bring a low overhead, efficient, multipurpose, repurpose-able, expandable, upgradable network architecture that can be integrated with existing networks. This heterogeneous network is the key to incorporating a unit level network into systems such as Blue Force Tracker and Force XXI and is the key to overcoming proprietary restrictions both in hardware and software.

### 1. Assumptions Made in this Research

There are several assumptions made with this work in order to focus on the networking component. As the Naval Postgraduate School seeks to deliver tactical squad enhancements through technologies, this thesis research focuses on the networking

element. The assumptions involve components of that network that are currently being researched by other projects within the same program. As stated before, this network needs some application running on it as a proof of concept for the data dissemination. For this purpose, the network is implemented for a squad of Marine riflemen. The network's purpose is to provide for data and communication capabilities within the squad. In order to do this, several elements of the system need to be integrated that are not a focus of this thesis.

An experiment was devised in conjunction with the other projects to test the network. A project being developed within the Reticle program at the Naval Postgraduate School to calculate whether a friendly-fire or fratricide condition exists serves as that test. The assumptions associated with this experiment are as follows.

The development of the program that determines these fratricide conditions is the first assumption. The inner workings and methods by which the program works are not a concern of the thesis; however, the program relies on the abilities provided by the network. In order to calculate fratricide conditions, the program requires the location data for members of the squad. The transmission of this localization data is the goal of this research. This localization data itself is not a focus and is the second assumption. For the purposes of this thesis, all localization data was assumed to be known. There is a project within the Reticle program that is currently working to solve the problem of indoor localization.

### 2. Projected Outcome

In order for this network to serve as a proof of concept, it must be able to integrate multiple platforms in both hardware and software. The components running the network must be selected in such a manner that it can be run on a low resource, low power machine. The smaller the processor and resources that are used, the smaller the device on which it can run. This allows for a lightweight product to be used by ground forces. The software must be simple enough that the overhead of the network does not slow down the traffic but robust enough to allow for further expansion. Upgradability must be built into the system so that it can handle new technologies, new data types, new message formats,

new applications and be reprogrammable to run across many different system types. This allows for computers and platforms with larger resources to also tap into the network and perform data analysis on whatever traffic happens to be sent across the network. The network needs to be able to cross any medium of transport from wired connection to wireless radio. This allows for the dissemination of data from a small wireless network of ground units, through a central tower, to an overhead drone, to a satellite uplink, and then anywhere in the world. There must be an element of security to the transmission and the data. One disadvantage of a network that connects everything together is that everything is connected together. If there is a compromise at one point of the system, it has the potential to compromise the entire system. Most essential, the network must be able to deliver data from the data producers to the data consumers.

The ability to detect friendly-fire situations is considered an application that runs on the network, like an application on a smartphone. That application, in order to work, needs a network that enables it to share data with its distributed components. The focus is on a solution for the network that carries the necessary information amongst its many application components. The network is required to integrate the different nodes in a scalable fashion, being able to run with as few as two nodes and as many as thirty nodes—at least in terms of software. When running with more nodes, it needs to be able to handle the bandwidth of data as well as deliver it quickly enough so that the indication of possible friendly-fire does not appear too late. The reaction time of the network is crucial for a smooth and useful product. Delayed dataflow can cause the friendly-fire indicator to trigger either too late or at incorrect times.

## C. COMPONENTS

The particular hardware, software, and architecture of the network that was created during the course of this thesis is discussed in this section. Several different components were used to make this system heterogeneous and interoperable between platforms, propagation techniques, and coding languages. These components are briefly reviewed and their role in the network defined. Additionally, the software that enables the network to distribute data is discussed and analyzed.

## 1. Arduino

The Arduino is a microcontroller that can be reprogramed and repurposed depending on need. It has inputs and outputs built into the board that support both digital and analog signals. As a hobby board, the software is open source and has become quite popular, requiring only a power source. The maximum operating clock speed is 16 MHz, which is not a limiting factor for the network at this point. The board is accessed and programed through a USB cable, allowing programs to be flashed to the onboard memory. The board does not run any form of operating system because it can only run the program currently flashed to memory. There is no ability to hold multiple programs and then select the desired one to run. The memory on board holds only one program, and it is overwritten during the next upload. There are many different versions of boards that the company sells, spanning across different computation and power levels. The company's most basic board, the Arduino Uno, was used for this thesis research and is shown in Figure 19.

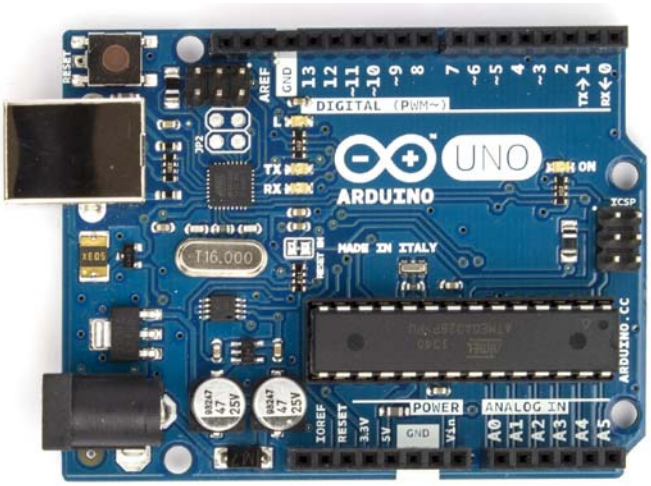


Figure 19. The Arduino UNO model that was used in this thesis, from [23].

For the purposes of this thesis research, the Arduino was used as a mobile node. The idea was to simulate the data that would be generated by each friendly node in the squad. Each Arduino based node was programed to create simulated data including position, orientation, and other data needed throughout the network. The Arduino was

able to simulate a physical node including transmission and power related dependencies. It played the role of a mobile, wireless node in the network. Many of the simpler, less complicated tasks within the network were carried out by the Arduino.

The Arduino connected to the network through the means of a ZigBee Xbee based wireless connection. Xbee radios use various different signal types to wirelessly transfer a serial bit stream of data. The devices come in pairs and are set up to take the place of a serial cable connection. Simply attach one Xbee to a port on the computer with a USB cable and the other to the device that needs to be wired, and there is a direct wireless connection between them. It essentially replaces a serial connection with a wireless one. The particular version of Xbees used does not have the firmware necessary to allow for mesh networking and is only wirelessly point-to-point connected to the computer running the network. Since the Xbee mesh network protocol is handled entirely by the hardware of the XBee chip, the implementation of a mesh network which only requires upgrading the chips. This is a possible future upgrade to the system which was not done during this research. An example of the Xbee Radio chip used is shown in Figure 20.


Figure 20. Example of a XBee Radio chip.

## 2. Raspberry Pi

The Raspberry Pi is a full System on Chip computer running a Linux operating system but in the package size of an Arduino. The computing power of the Raspberry Pi is less than that of a full sized desktop processor but more than the Arduino. It is about the size of a modern cell phone with ports for power, visual output through HDMI, and USB ports. A USB keyboard and mouse can be attached, and the entire thing boots up to a partial Linux build, complete with graphical desktop interface. There is also an Ethernet port which was used extensively in this research, both for access to software online and to network the devices together. The fact that the Raspberry Pi operates as a fully functioning desktop computer allows for easy interfacing and prototyping. The difference between this and the Arduino is in how the two components interface, store, and execute programs. On the Arduino, it is a slow process to essentially install a new program, overwriting the old one, every time. With the Raspberry Pi, the program can be written and compiled on the same device. The advantage of the Raspberry Pi is that the process is more like a PC. The Raspberry Pi allows the user to download and install additional software without having to replace the existing software. Programs can also be downloaded straight from the Internet onto the device. Instead of housing a single program at a time in flash memory, the Raspberry Pi can carry multiple programs in an expandable SD memory card that plugs straight into the device.

A very useful feature of the Linux based systems is the ability to secure shell remote login into each device. With this feature enabled, if all the Raspberry Pis are on the same network, it is not necessary to have a monitor and keyboard for each one. Instead, from one of them, a secure shell can be created that connects the keyboard, mouse, and monitor to the remotely located Raspberry Pi. It allows for the use of one keyboard and monitor to control the whole network of devices. This is very useful because when parameters needed to be changed, there was no need to unplug a node and re-flash a program too it. When a change is necessary, simply secure shell into the particular device and change or replace programs and data. New files and programs can even be pushed across the network to all devices in order to add new capability to the nodes on the network. This saves time once all the initial overhead to set up the network

is complete. All that is required to create a secure shell connection is the IP address of the device, the username, and the password.


Figure 21. The Raspberry Pi B+ model, from [24].

For the purposes of this research the Raspberry Pi, shown in Figure 21, served as the next evolution of network nodes. With the power and ability of the Raspberry PI, the node became more like the ideal final product. Where the Arduino was simply a location beacon—the Raspberry Pi has the ability to run multiple programs and other applications in addition to broadcasting data to the network. Since the Arduinos had shown versatility and the ability to used wireless connections, the Raspberry Pis were chosen to be a wired connection. Networking on the Raspberry Pis consists of a wired network through a router. With built-in Ethernet ports, rapid prototyping of the network is much faster. It also shows heterogeneous functionality with the Xbees and Arduinos. Moving to a Wi-Fi based network by connecting them all to a single access point with USB Wi-Fi dongles is a simple next step.

Throughout the testing, the Raspberry Pi proved capable of running the entire network. Previously, the network core was run from a desktop computer, but a trimmed down version of it is able to run on the Raspberry Pi. Currently, the applications running on the Raspberry Pi are written in Python. If necessary, the Raspberry Pi can also run other programming languages, most popularly C++. Ideally, the network workhorse

remains the Raspberry Pi because of its ease of use and open source nature; however, there are other embedded Linux platforms that are smaller and may also be incorporated into the network. Particularly, adding Linux based Gumstix boards would reduce the size and weight by about half. Currently, the size and weight is already very small, but if the need were to arise for even smaller platforms—Gumstix would be ideal.

### 3. Robot Operating System

The Robot Operation System (ROS) is an open source program that is an operating system for robotic systems. It is very prominent in universities and research labs. Recently, it has begun to move into the commercial sector to operate robotic systems for production. The Department of Defense has also recently begun to use ROS for military robotics programs. The organization's most recent claim to fame is the implementation of ROS on a robotic system currently operating on the International Space Station. ROS serves a crucial role in this research and is more thoroughly explained in Chapter IV, Section E.

## D. LAYOUT OF THE NETWORK

With the mix of components discussed, the interoperability of the network is demonstrated. Not only are these components physically different, but they use different coding languages, different mediums of communication to the network, and serve different roles in the network. The Arduinos and Raspberry Pis are used as the mobile nodes for the individual marines. This configuration is shown in Figure 22. The Raspberry Pi is on the left with a CAT5 cable connecting the Raspberry Pi to the desktop computer. On the far right is an Arduino. The blue chip on it is the Xbee radio that wirelessly connects it to the desktop computer via the computer's own Xbee chip.

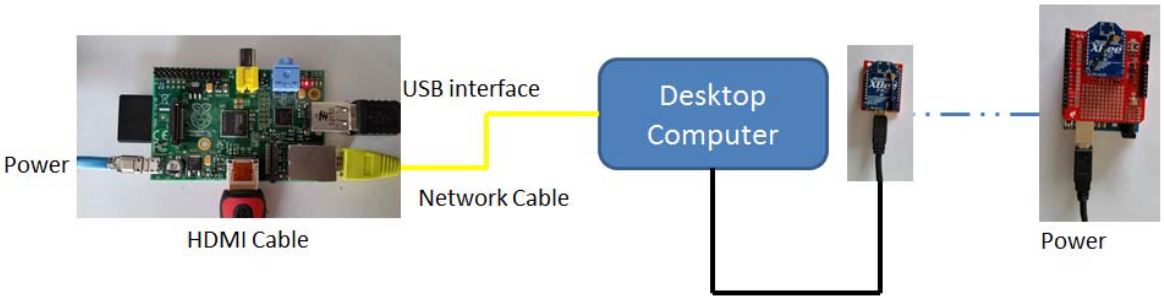


Figure 22. Example of how the components were connected together.

There is an alternative method of connecting the network in which the desktop computer was replaced with another Raspberry Pi. The build shown in Figure 22 is an early build to enable easier prototyping and testing. This enabled faster computing and visualization of the data. The choice to use both wired and wireless connections is purposeful. This was done to prove the ability of mixed communications in the network.

## E. ROBOT OPERATING SYSTEM

The abilities and functions of the ROS are the single most influential component to the entire network. More than any other component, this software makes the network unique.

This software was originally designed as an open source operating system for robots. With the robotic market expanding exponentially, a large problem of redundancy arose in the community. Universities and research departments were each recreating elements of software from the ground up. Each piece was proprietary to the university and the hardware of the robot—even if they were all using the same hardware. Motor driving software was built specifically for each robot and motor combination. In order to overcome this single-use coding problem, ROS seeks to make the interfacing of these components easier. ROS changes single-use code by encasing the software elements of a robot into black box subsystems. It is a slowly growing library of these functions that given the correct inputs, give the desired outputs. For example, an object recognition program essentially needs only a video input. With that video input the software can analyze the environment and report the distance and relative angle of the objects. That position data is considered the output of that block. That data can then be given to a

navigation block of software that takes two inputs, positions of objects to avoid and the programed destination for the mission. Those two inputs give an output to the motors to drive the robot. With ROS it is only a matter of connecting these blocks together to create a working program.

Another way to look at it is that software components are treated similarly to the physical components. Sensors only deliver raw electrical signals, and software interprets these signals into data. In the past a program for a robot has been one large piece of code that interprets signals and runs the higher level logic algorithms all in one place. ROS has modularized this approach. It breaks the larger code into pieces that become building blocks of the system. These blocks are their own individual systems that require certain inputs and deliver certain outputs. All that is required is to connect the system correctly. With ROS, software is just as vital as hardware and is treated in the same physical way. What makes this so unique is that when that software component is no longer desired, the user simply unplugs it and plugs in a new one.

ROS enables the robot designer to view the system at a much higher level than it did before. Instead of having to build object recognition software for each robot, simply connect the camera to the recognition block, that block to the navigation suite, and finally the motors. Ideally, the experience of building robots is now easier and more accessible. ROS's functionality brings large-scale projects down to a more manageable level and enables robot developers to pick these modules and put them together in unique ways. With an open source library constantly being updated, the possibilities are endless.

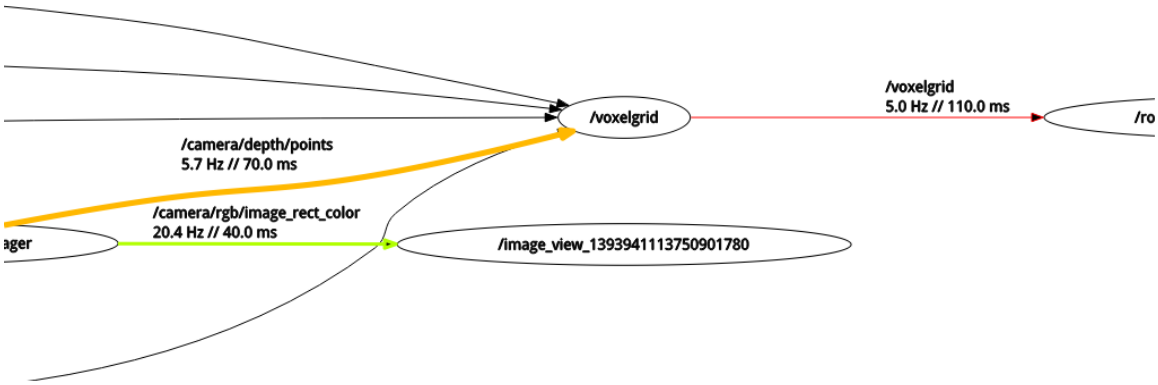


Figure 23. Example of ROS nodes with topics connecting them, from [25].

Individual components called ROS nodes, which are connected by the signals called topics sent between them, are shown in Figure 23. The topics are the arrows, and they show the refresh rates of the individual signals called messages. A camera is sending images to an image viewing program node at 20.4 Hz and sending another version of the image to a grid analyzer at 5.7 Hz. Throughout the rest of the paper, the components and devices are called nodes. Nodes either produce messages by publishing or use messages by subscribing. The messages sent by individual nodes and are delivered through topics that are collections of certain message types. The topics are the objects to which nodes publish and subscribe. Multiple nodes can publish and subscribe to topics at the same time.

The network built for this research uses ROS uniquely as a message broker. The typical architecture of a robot using ROS is a vertical hierarchy of systems. Each component of the robotic unit is a cog in one larger machine. There is a hierarchy and an overall goal of the single device or robot. Components of the system are subordinate to another component. This expected layout of ROS is turned on its side and used horizontally for this network. This is shown in Figure 24. There are the occasional vertical components, but because of the versatility of the network, that dominant node can change based on the mission. Now, instead of gathering sensor data for the robot's purpose, the network gathers and distributes data throughout the network for the purpose of every node. Different nodes can come online and utilize the data in different ways, for different purposes.

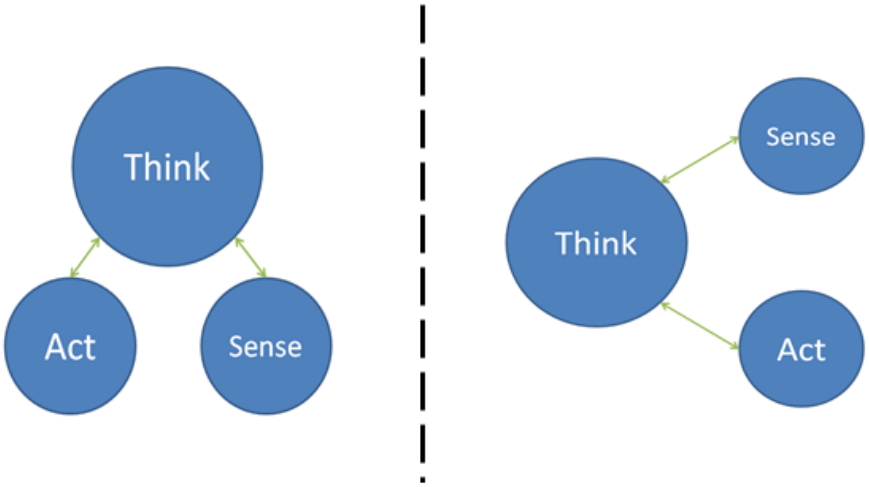


Figure 24. Typical architecture of a ROS program or network versus how it was implemented for this research.

To consider a different example, the basic idea that ROS uses can be explained as a universal language. Much of Europe may not speak the same language, but they all use Arabic numerals. Having a common numerical system gives a common thread of communication through which information and knowledge can be passed. Those countries can share proofs and communicate knowledge to each other through mathematics without having to understand the same spoken language. More importantly, that knowledge can be used by the other countries, and they never have to understand what the originator of the information was actually saying in his or her language. They need only receive the mathematical data. The Pythagorean Theorem is certainly used today, but not everyone speaks Greek.

In order to act as this common language, ROS is added to the code of a program to act as the interface between the software and the network. The overhead that ROS has in a program is typically at the top of an existing code. It declares the particular preformatted message types that are used within the network. These can be simple message types like strings and integers or more complicated ones like arrays or string-array combos. The network message formats are then mapped to variables within the program. As the program runs and updates its variables the ROS message grabs those variables, packages them in to the preformatted message, and sends it to the *roscore*

program under a particular topic name. The *roscore* program is the orchestrator of the network and serves as the data broker.

This general idea is shown in Figure 25, which represents a local program running a loop and producing some changing variable. In the hashed lines above is a representation of the ROS code that is added to this program. It grabs the changing local variables of the program and copies them into ROS variables that are used to create a message that is sent across the network. The message format shown in Figure 26 is an example of this format in greater detail. For the sake of this example, the message name is called 'Update' and contains all the variables listed in the box below it. Those variables are filled with data that is being produced by the local program loop. This demonstrates how a message can carry many variables within it. They do not all need to be used but are available to anyone subscribed to the topic that to which the messages are posted.

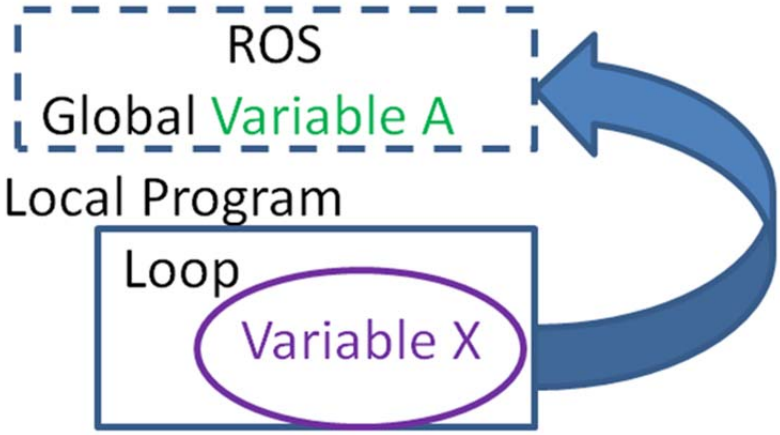


Figure 25. ROS adds code to the top of programs to map local variables to ROS messages.

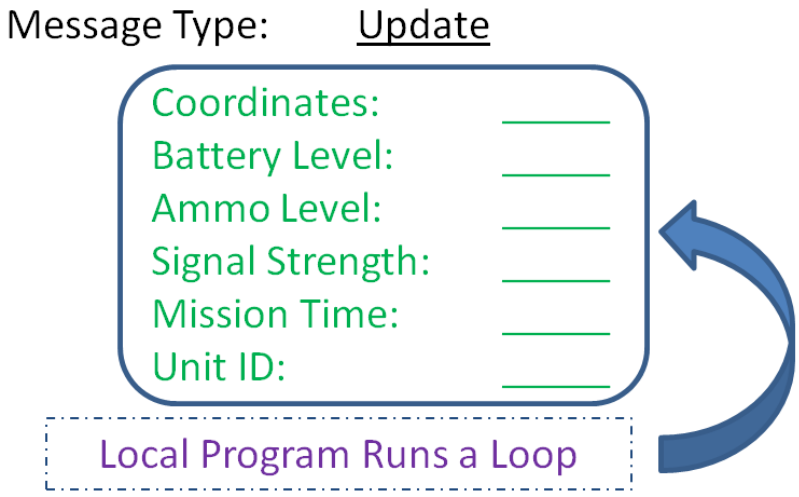


Figure 26. An in depth example of potential ROS message content.

Any other component of the network that is listening to a topic then receives the formatted message from the Broker and withdraws what that program needs. How nodes can publish data to a topic and how separate nodes can subscribe to that topic and receive the messages posted to it is demonstrated in Figure 27 and Figure 28. This again reiterates how the data consumers are unaware of the data producers. The second example shows the bidirectional abilities of the data flow. A node can both publish and subscribe at the same time, either to the same topic or to different topics. What is not shown in these figures is the data broker. For simplicity the broker was left out but would be the entity setting up the publishing, subscribing, and topic. The connections through which data flow on the network are established by the data broker.

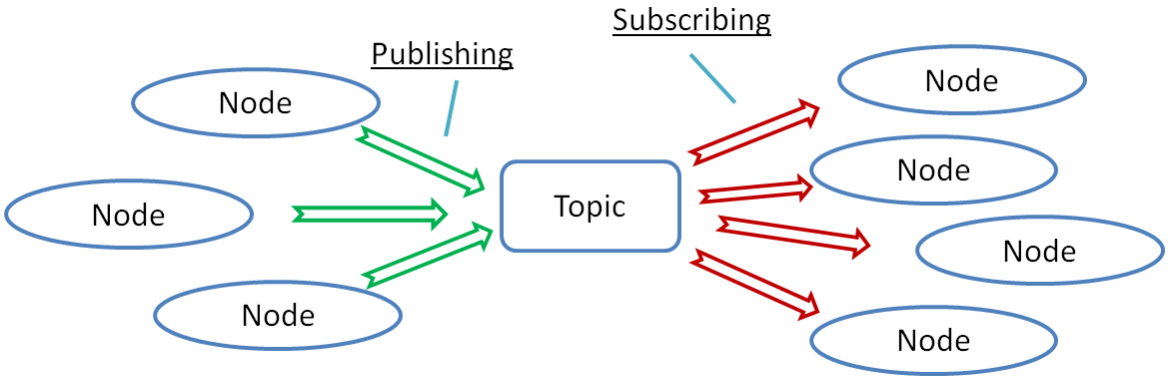


Figure 27. An example of multi-connection topics.

It is from this architecture that ROS derives its cross-platform compatibility. Like integrating different components on a robot, these different hardware components are able to be incorporated into the network. With the preformatted message types, all these components now speak the same language on the network regardless of the programming languages used in each node. With a common message format, the messages are readable by the message broker and every other node. Each message goes within its designated topic both for quick reception of data and quick delivery. Remember that ROS's original design is to allow sensors to deliver data to the network, software nodes to analyze it, and actuator nodes to react to it. This quick turnaround of data is the exact attribute desired in the planned network.

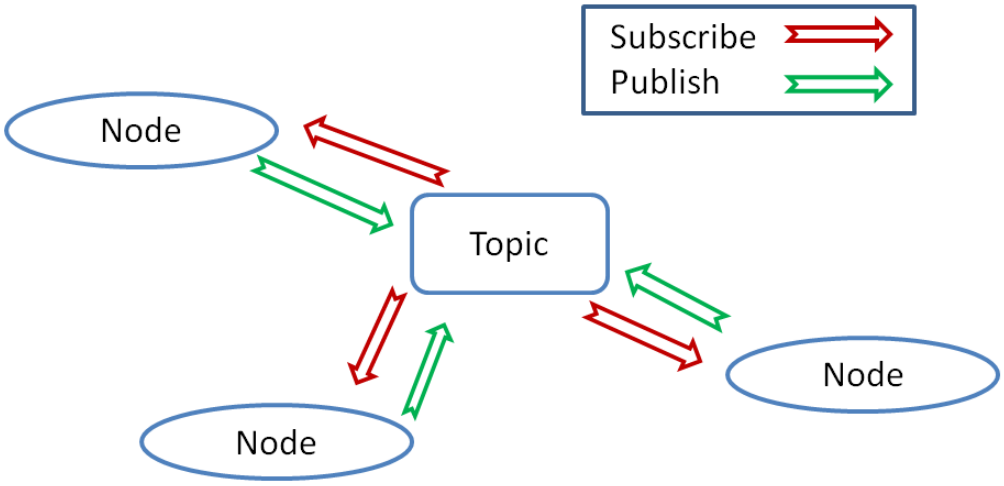


Figure 28. An example of bidirectional data flow in a ROS network.

The use of a data broker scheme changes the design of the network. In typical network architecture, the data would be sent to a specific address of a host on the network; however, in the architecture presented in this research, there is no addressing of the data. Data is simply sent to the broker who manages its flow to those who need it. Figure 29 is an example of what the messages look like going through a topic called "Updates." The command "*rostopic echo*" is used to have the message broker echo any messages that flow through the topic "Updates." Through the message broker, the messages can be accessed manually in this manner through a terminal window. That functionality was used extensively to view the message traffic through the network. This is groundbreaking to prototyping, experimentation, and research. It is extremely useful in rapidly prototyping the network built for this thesis research. The ability to rebuild the network in any way desirable is the key to its versatility.

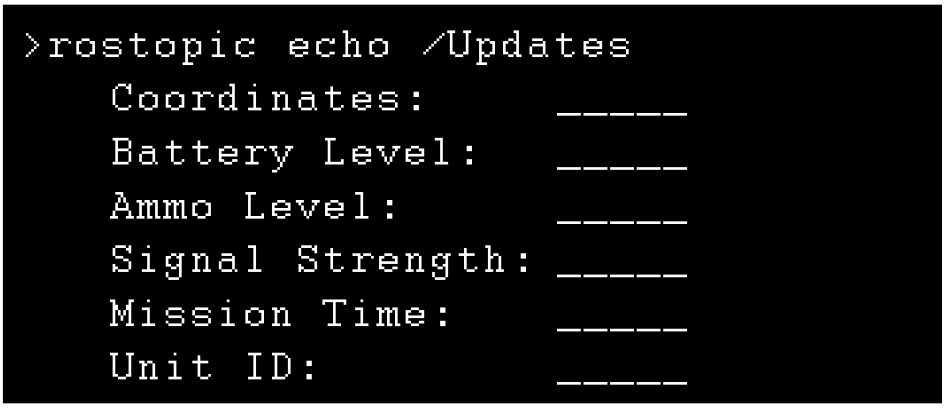


Figure 29. Example of how messages are viewed by the data broker.

## F. UNIQUENESS OF NETWORK

To the best knowledge of the author, ROS has never been used as a network data broker before. The typical top down infrastructure has been turned on its side. The data of the network does not go through one central point (the robotic brain); it now flows horizontally throughout the network. The ability to integrate different hardware components into one robot (the original purpose of ROS) is now used to implement different hardware components into a network.

Given the current setup of the network, there is essentially a computer on each soldier now. The Raspberry Pi has four USB ports for expansion. Simply plug in a new sensor to use and add the coding to the memory of the Raspberry Pi. The network infrastructure is now there to support expansion in the types and formats of data sent through it. The nodes themselves are ready for expansion as well through the extra available computer power. There is no further addition of equipment needed besides the sensor components which are typically lightweight and the software, which adds no weight.

## G. THE KILLER APP

With the network layout established, there needed to be a method in which to test its abilities. A friendly-fire detection system was created for another project at Naval Postgraduate School. The goal is to use this friendly-fire detection system as a test for the network. The network needs to be able to provide the backbone of shared communications among the devices in order for them to work properly.

The program uses a motion capture sensor to capture the motion of the rifle and then a microprocessor to compute the movement of that rifle with the movements of others in the squad. The Arduino Unos do not have USB ports and are used only as location beacons for friendly nodes. They can be thought of as noncombatants in a practical scenario. The Raspberry Pis represent the Marine with a rifle. Each one has an inertial motion capture sensor plugged into the USB port. They track both the position of the Marines and the pose of the rifles. One of the Raspberry Pis acts as the squad leader and carries the *roscore/data broker* program. There are no additional physical characteristics associated with the node carrying the *roscore*, only a software change. Any of the nodes can potentially serve as the *roscore* if needed. The network delivers the locations of those other members in the squad. The detection algorithm is reliant on the location data that the network is able to deliver. The sensor used was a YEI Technologies sensor package depicted in Figure 30. The microprocessor was the Raspberry Pi. The program computing the algorithm is written in Python, a simple terminal programming language. At the top of that script was the integration to ROS and the rest of the network.

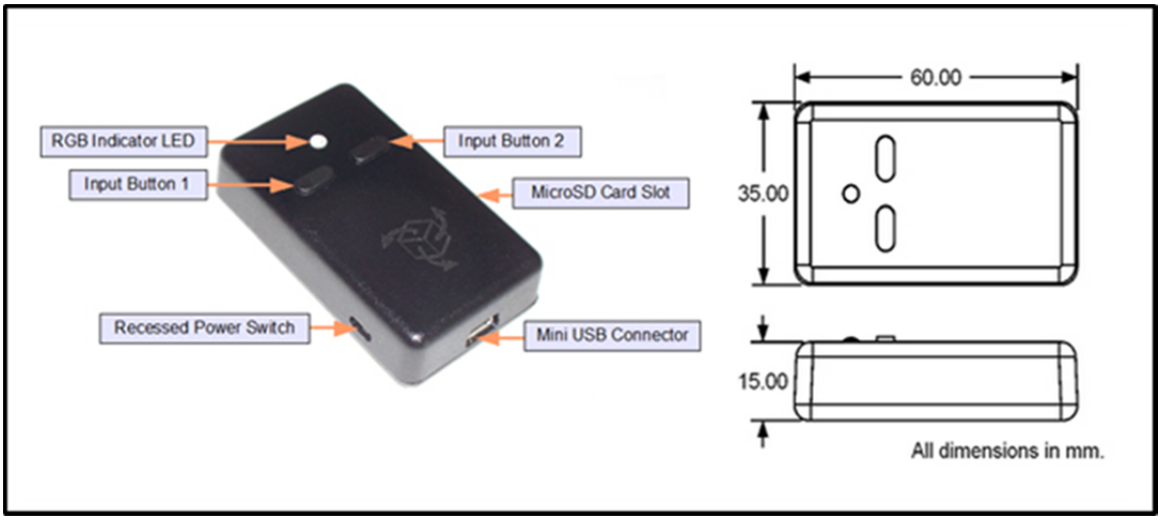


Figure 30. The 3-Space Data-Logging sensor from YEI Technology was used as the motion tracking sensor, from [26].

The goal of the motion capture system was to integrate it into the Marine's load-out without adding significant weight. The ideal final setup is shown in Figure 31. Currently, the added weight is less than 1 lb without a power supply, but it can be seen that the system integrates well with the current load out. Because the network is currently running on WiFi, Xbee, and Raspberry Pi, one only needs to add the program and sensors needed for the new functionality. To integrate this application into the network, the YEI sensor and the code to use the sensor was added. The Raspberry Pi and network interface were already running the network. This level of integration can continue into the future. To add a camera, simply plug it in and write the code. The rest of the hardware is already present.

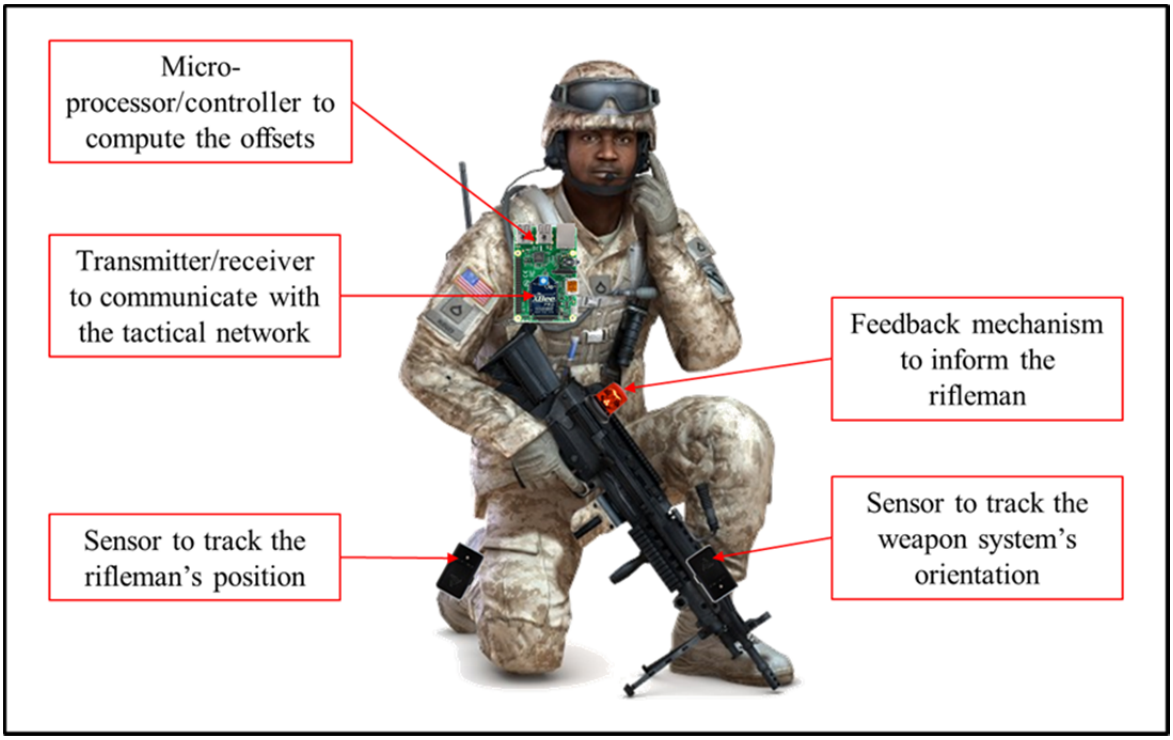


Figure 31. Example of the sensor package carried by the Marine, from [26].

## H. DESIGN CONSIDERATIONS

Several factors were considered with this scenario in order to design a network that can provide data fast enough. Because of the flexibility with a ROS implemented network, the choices in hardware and equipment came down to the application. Since the Arduinos serve as a position beacon and have no need of high level computation, they can remain a simple platform that consumes less power. Other features of the network had to be more critically analyzed.

An example of how the program works is shown in Figure 32. The cones represent the field of fire for each Marine. Each is measure 8.5 degrees left and right of the barrel. The green color indicates there are no conflicts, while the red represents a conflict. In this example there are three Marines in a building, all in separate rooms, but able to determine whether their friendly units are on the other side of the wall. This is useful in situations where the walls may be very thin and allow a bullet to pass through.

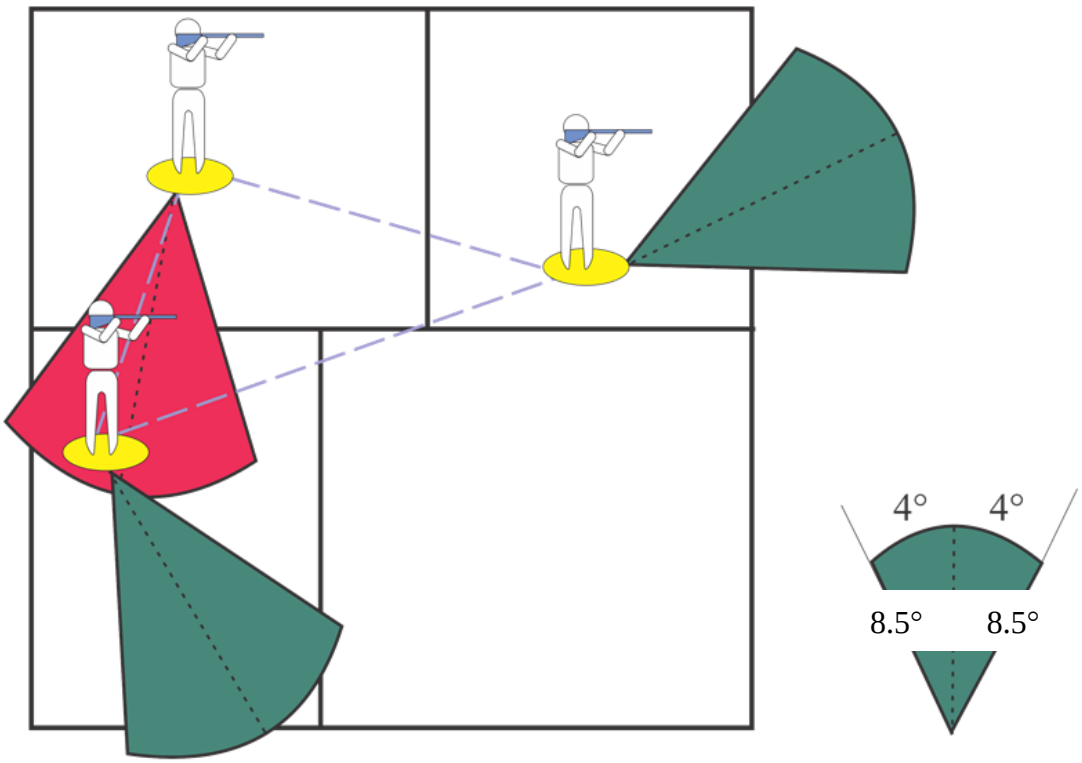


Figure 32. How the fratricide application works.

The latency in the network is a primary concern for this application. Data transmission needs to be fast enough so that as the rifle swings through an arc to clear a room, the network can keep up with position update demands. The idea is that as a rifle is swung through certain areas to clear a room, the data delivery needs to meet or exceed the speed of the sweep. To decide how fast the network needed to update the positions, two things were considered: the speed of the Marine and the speed of the rifle sweep. Depending on the range, we see that the rifle sweep can draw out a much larger area of potential error than the movement of a Marine on foot. With that in mind, the first consideration was how fast the rifle sweep may be and how far away the Marine may be. Distances between 5 and 20 meters were selected for testing based on the scenario being indoors. Rifle sweep speeds of between 0.5 rad/s and 120 rad/s were considered based on interviews with Marines.

Several calculations were run and the maximum and minimum values are shown in Figures 33 and 34. The typical distance for the given scenario is 10 m. With a sweep rate of 30 rad/s, any frequency above 10 Hz gave an acceptable margin of error of 3 m. The hashed vertical line highlights the values at 20 Hz, the chosen frequency, and the horizontal solid line represents the acceptable error of 3 m. Figures 33 and 34 were plotted using

$$Error = \sin\left(\omega_{Network}\phi_{Marine}\right)\Delta_{Target} \tag{1}$$

where the refresh frequency is represented by $\omega_{Network}$ in units of Hz and the angular speed of the Marine's sweep rate is represented by $\phi_{Marine}$ in units of radians per second. The distance of the target is represented by $\Delta_{Target}$ in meters.

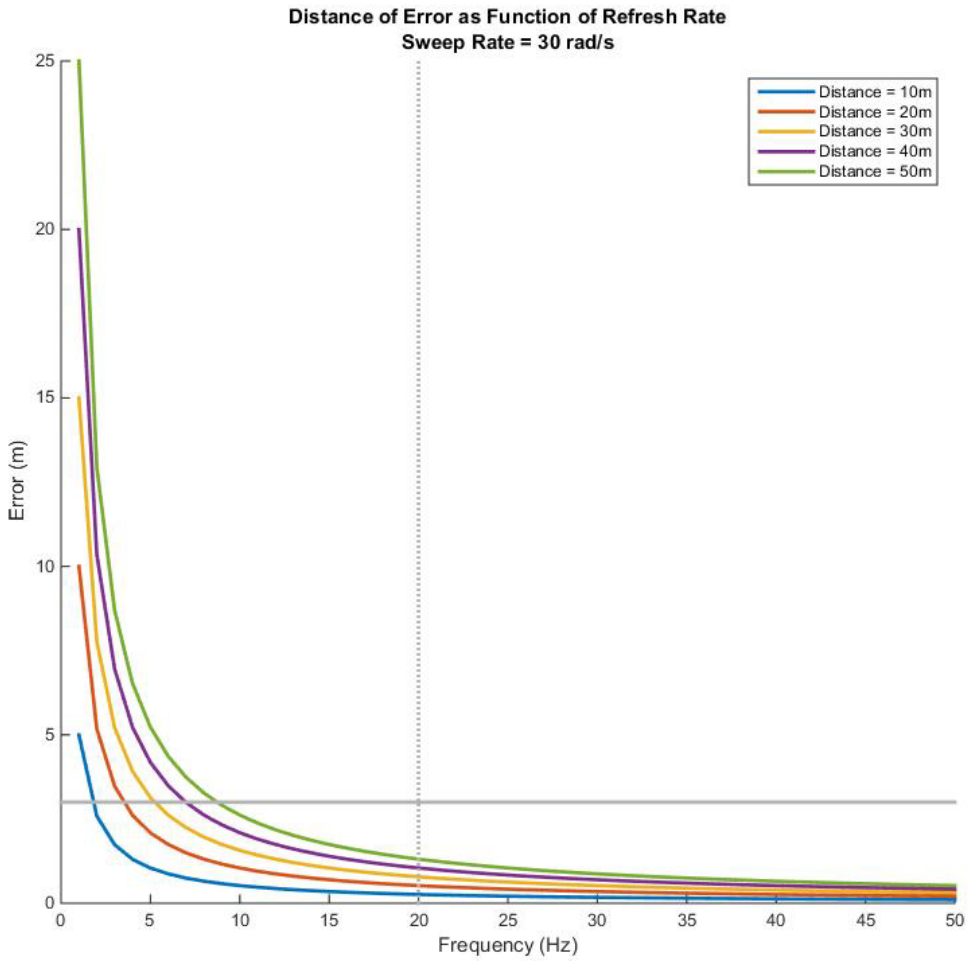


Figure 33. Computed error rates at different distances and frequencies for a sweep rate of 30 rad/s.

The same calculations made with a rifle sweep rate of 120 rad/s are shown in Figure 34. The error stays below our threshold of 3 m until a distance of 30 m from the Marine. This is acceptable given our current scenario, but the refresh rate should be scaled based on distance of the target al.l of these errors are calculated assuming the worst possible conditions and represent the maximum error value possible. They do not represent an average error. With that in mind 20 Hz is acceptable for the scenario.

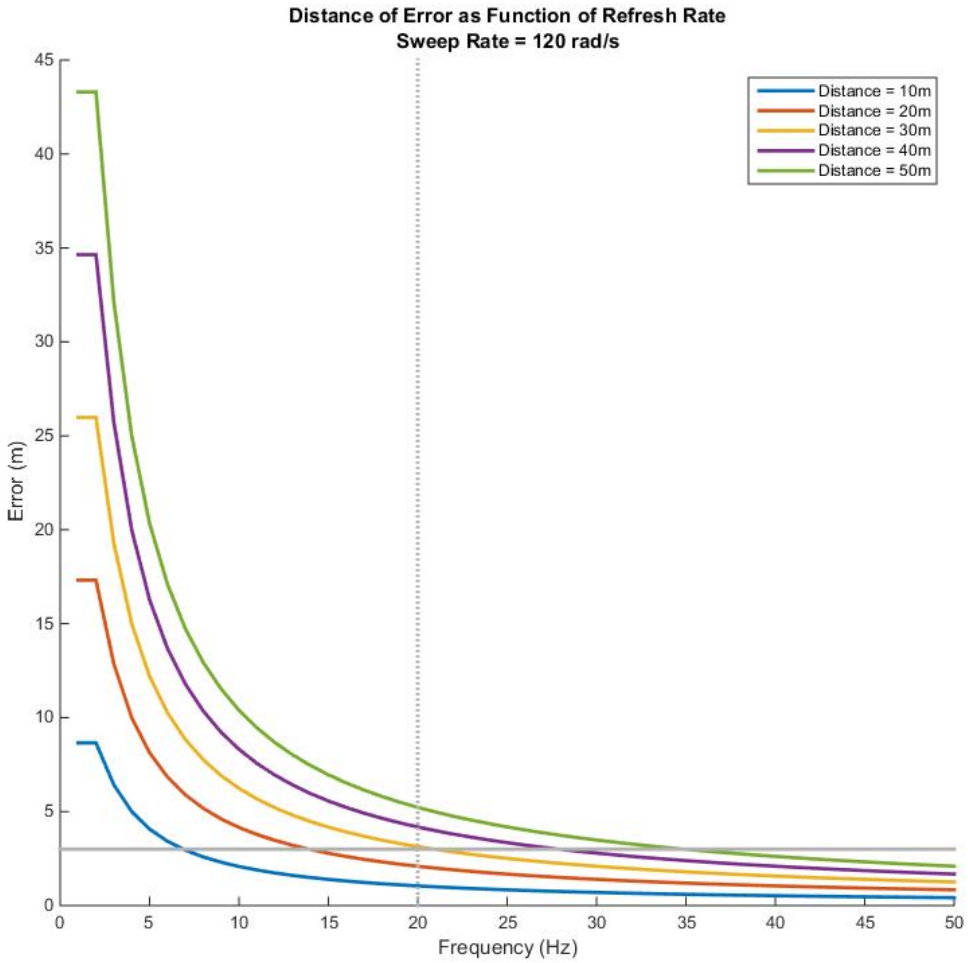


Figure 34. Computed error rates at different distances and frequencies for a sweep rate of 120 rad/s.

A consideration with the range of the target is the speed with which it is moving. Based on the speed over ground measurements, the refresh rate of the network may show the individual jumping a certain distance at a time. To determine the refresh rate necessary for good resolution of movement, the maximum and minimum speeds calculated for this scenario are shown Figures 35 and 36. These figures were computed using

$$\text{Resolution} = \left(\mu_{Marine}\omega_{Network}\right) \tag{2}$$

where the resolution of the network is computed by looking at the speed over ground of the Marine, represented by $\mu_{Marine}$ in meters per second to the refresh frequency of the network $\omega_{Network}$ in Hz.

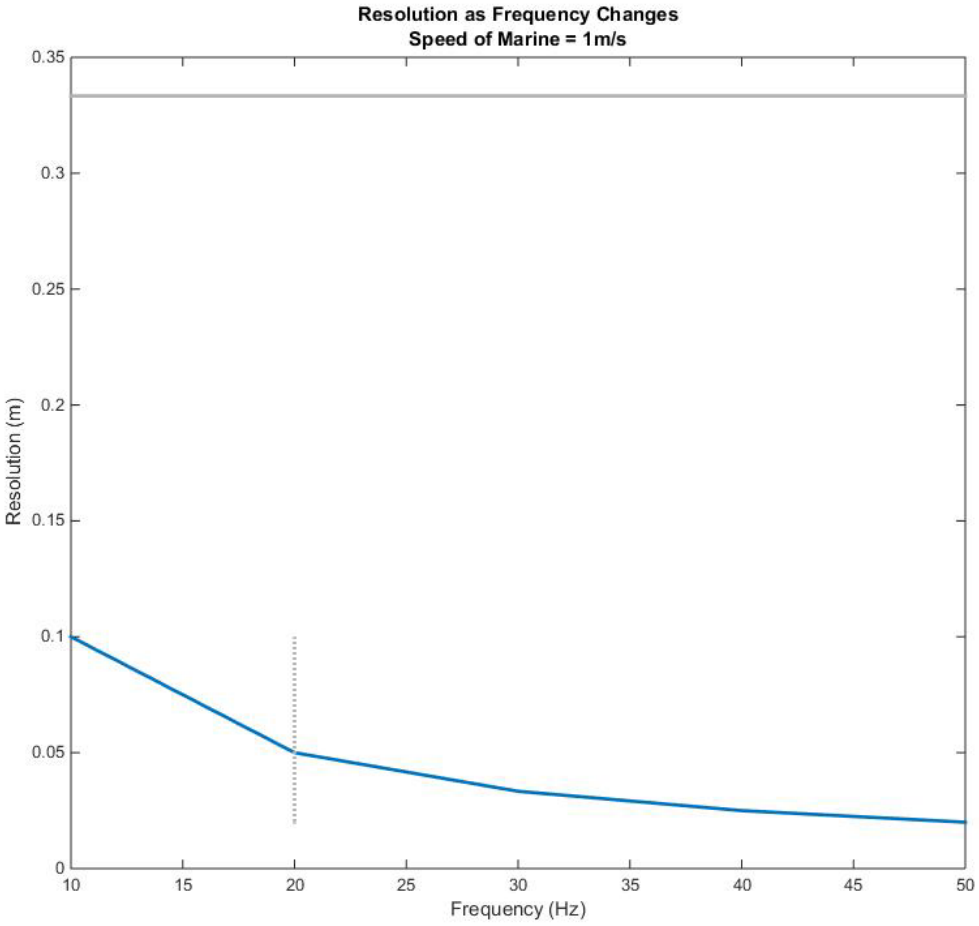


Figure 35. Computed resolution of node's location based on frequency of position updates.

The resolution threshold was set at 0.3 m in order to reduce false positives in the detection algorithm. The grey horizontal line represents the threshold of 0.3 m. The results suggested that for the scenario, anything above 10 Hz was acceptable. Since the limiting factor is then the sweep speed, the choice was to use 20 Hz. Figure 35 was calculated using a speed of 1 m/s for the movement over ground and Figure 36 uses 6 m/s over ground.

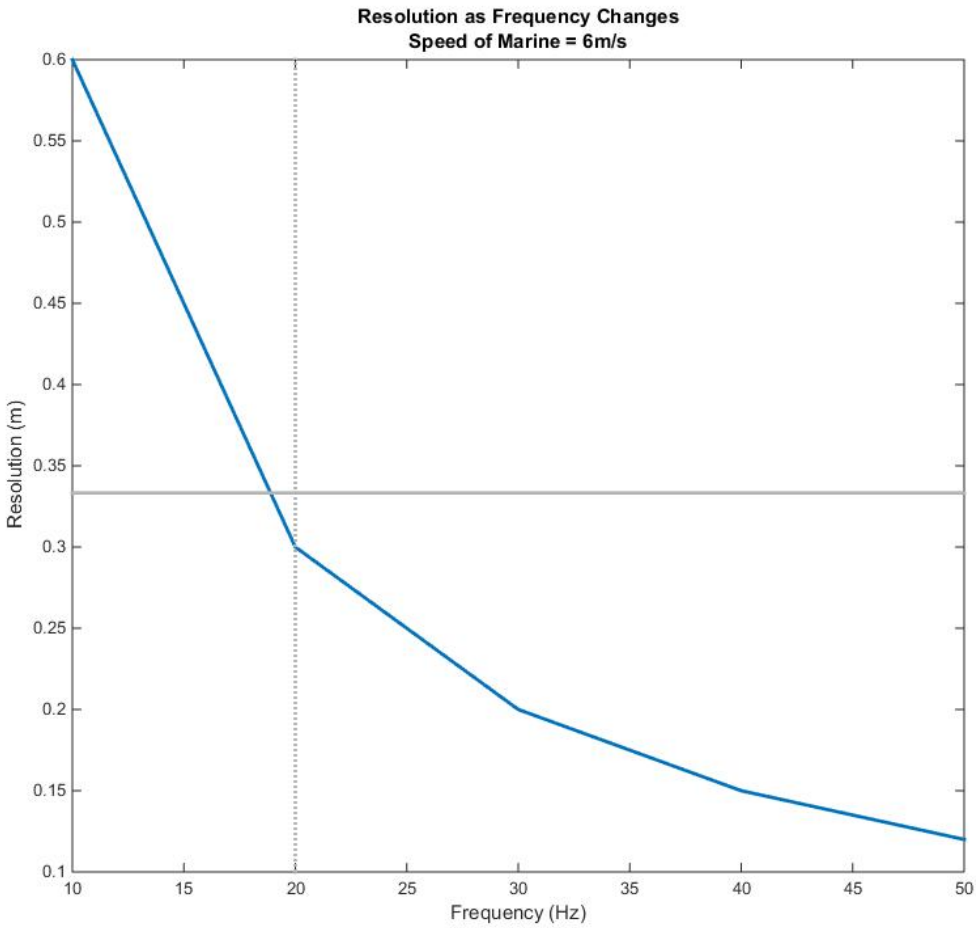


Figure 36. Computed resolution of node's location based on frequency of position updates.

The final consideration for the scenario was the throughput with respect to the number of users on the network. The network must be able to hold at the very least ten users for the squad. Ideally, for future implementations, the ability to hold hundreds is preferred. The bandwidth consumed depends on the message sizes, the number of users, and the number of message from each user. Message size is based on using the average message size for ROS message of 9 kb. For the given test scenario, the message size is 4 kb. The average value was used to determine the networks limitations in order to have a margin of error and the ability to expand the network in the future. The limiting factor in the network is the wireless connection with the Xbees which transmits at 9600 kbps. The number of users possible on the network given three different frequencies, 10 Hz, 20 Hz,

and, 50 Hz are shown in Figures 37, 38, and 39, respectively. The grey horizontal line represents the maximum baudrate. These figures are computed with

$$\text{Network Load} = \left(2N_{Users} X_{Size} \omega_{Network}\right) \tag{3}$$

where the load on the network is determined by the number of users represented by $N_{Users}$. The number of users is multiplied by 2 in order to account for the scenario of sending a message and receiving a message. This assumes the worst case loading scenario of sending and receiving at the same time. The number of users is multiplied by $X_{Size}$ in kilobytes, which is the size of each message packet and then multiplied by the frequency of the message being posted to the network, again represented by $\omega_{Network}$ in Hz.

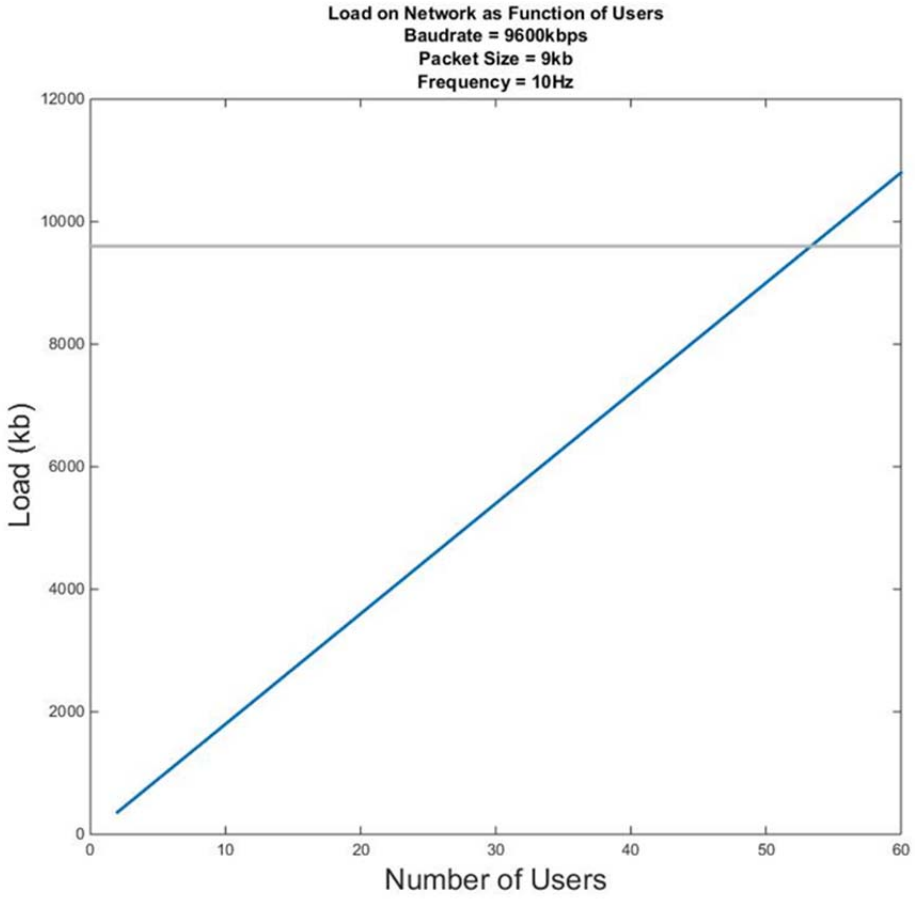


Figure 37. Computed number of users possible based on message size at 10 Hz.

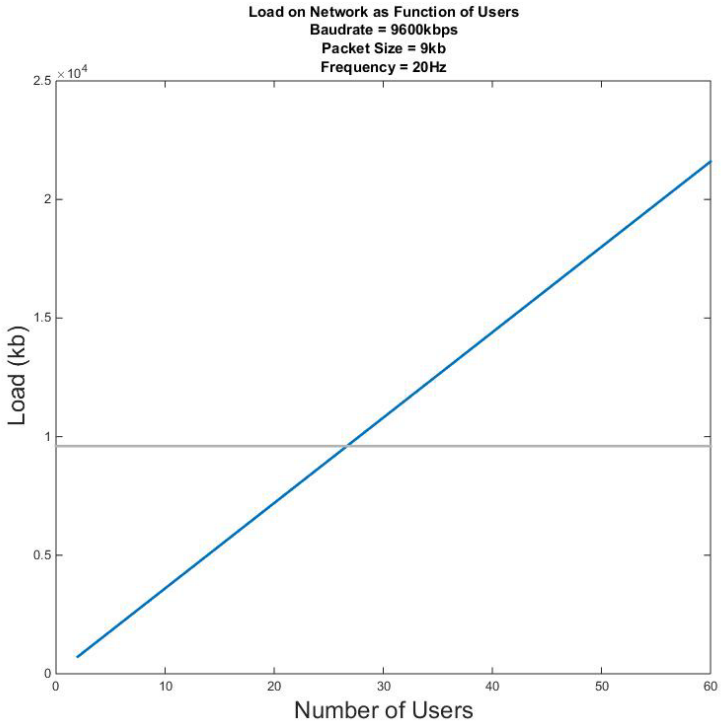


Figure 38. Computed number of users possible based on message size at 20 Hz.

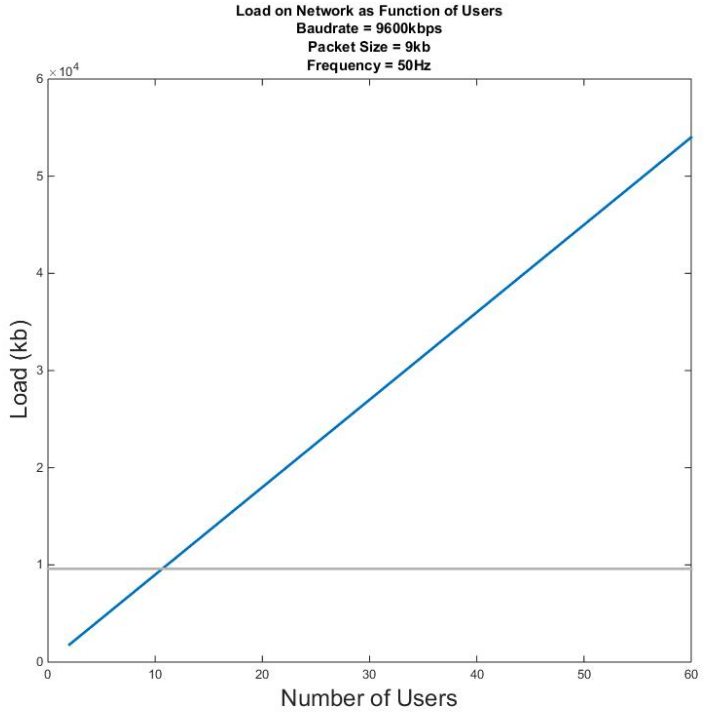


Figure 39. Computed number of users possible based on message size at 50 Hz.

At 20 Hz, as can be seen in Figure 38, shows the network holds nearly 30 users, which is well over the ten required for this scenario. The challenges of changing variables and how they affect the network are shown in Figures 37 and 39. Many of these can be overcome with modifications to the baud rate, message size, and frequency. This analysis also assumes the worst case scenario of having every node on at the same time and transmitting at the same time; however, for the scenario, the chosen equipment is well within the operating conditions necessary to achieve accurate readings of where all nodes are located.

## I. INTEGRATING THE FIRST NODES

For the first experiment, the central *roscore* is located on a small portable computer. The advantage of this device in the early testing phases is the large amount of RAM and high processing speed. With this robust device, running high graphical output tests is much easier than with a low RAM device. Many of the early tests involved having a lot of visual outputs to the screen for confirmation of working code.

For the purposes of this experiment, the positions of the devices were either fed in remotely or preprogrammed onto the devices. Each device consists of the ability to determine its rifle pose and broadcast its position. With shared location coordinates stored by the broker, the rifle nodes pull down each other's positions from the broker and compute whether there is a conflict of fire. The data flow in the network is illustrated in Figure 40. All of the positions are stored in a topic the data broker calls 'NodePositions.'

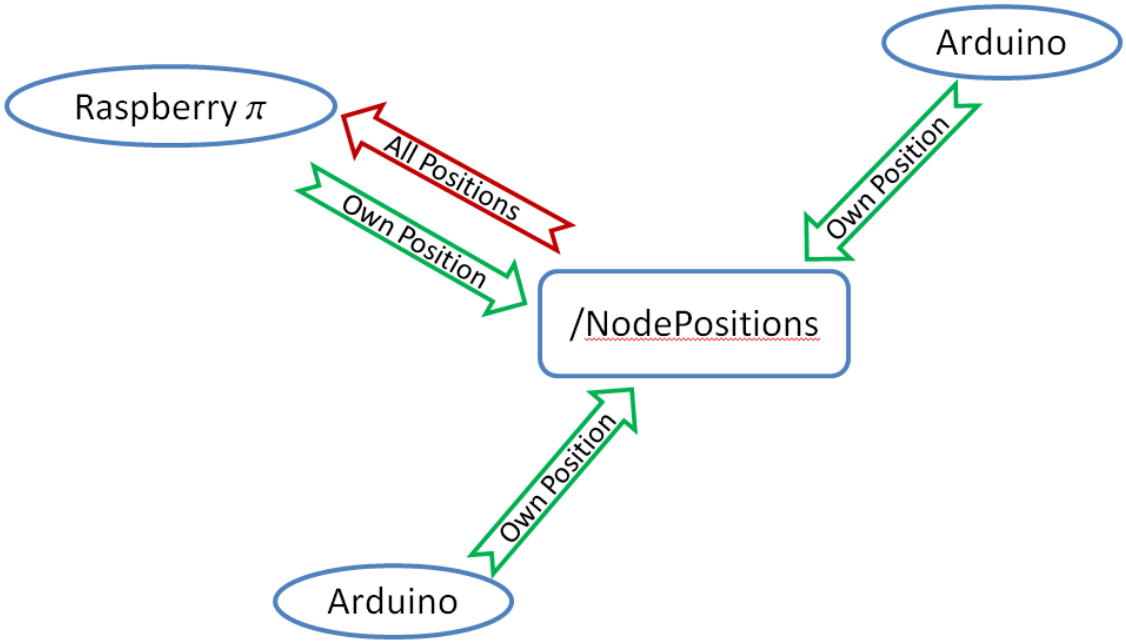


Figure 40. Illustration of data flow through the network.

First is the use of the Arduino as a positon beacon only. It does not have the rifle pose algorithm on it but simply broadcasts its position on the network. The first experiment includes several Arduino boards using Xbee wireless connections to connect back to the computer. These Arduino position beacons were placed throughout the lab. Each one needs to know its own position, which was accomplished by placing them in known geographical locations in the lab and then programming those positions into the devices. In the first phases, the measurements were kept to the axis of the rifles frame. This way the rifle can be pointed straight ahead, down its axis, to determine detection. If the friendly node is right in front of it, the warning goes off. To test safe conditions, simply rotate the rifle eight degrees off of the axis, and the warning goes away. The positons are broadcast in a three coordinate format, [*x*, *y*, *z*]. The rifle is programed to align its axis to the Northeast Down coordinate frame.

## J. THE CODE EXPLAINED

In this section, some time is taken to break down elements of the code to further explain how ROS integrated with the conflict algorithm code. The basic breakdown of software elements within the network is shown in Figure 41. Each rifleman node essentially has two components. Both components are currently part of the same program, but each serves a distinct role. First is the part of the program that assesses the current position of the rifleman. As stated earlier, for the sake of this research, the ability to accurately determine the location of the riflemen is assumed possible. In the tests, these location coordinates were supplied from external sources. The second component of the program takes in locations of friendly riflemen and computes whether one of those positions lies in the field of fire of the individual's rifle. If so, a warning is issued to the person holding the rifle that a conflict exists. There are no warning messages sent over the network, only position coordinates. The red lines depict the positions of the riflemen being sent to the data broker in a three coordinate format. Each red line is a single set of coordinates originating from the riflemen. The blue lines represent the constant retrieval of the locations of all riflemen. These are retrieved in a first come, first serve manner from the data broker. As each location is received by the conflict algorithm, it is stored in an array or matrix within the program to be computed.

There is a small difference in the squad leader's equipment. There is no physical difference in his equipment, but the data broker resides on his Raspberry Pi. Additionally, the blue block in the middle represents the various communication mediums used in the project. Because of how the system was designed, the physical medium does not matter and, as such, various different mediums were used.

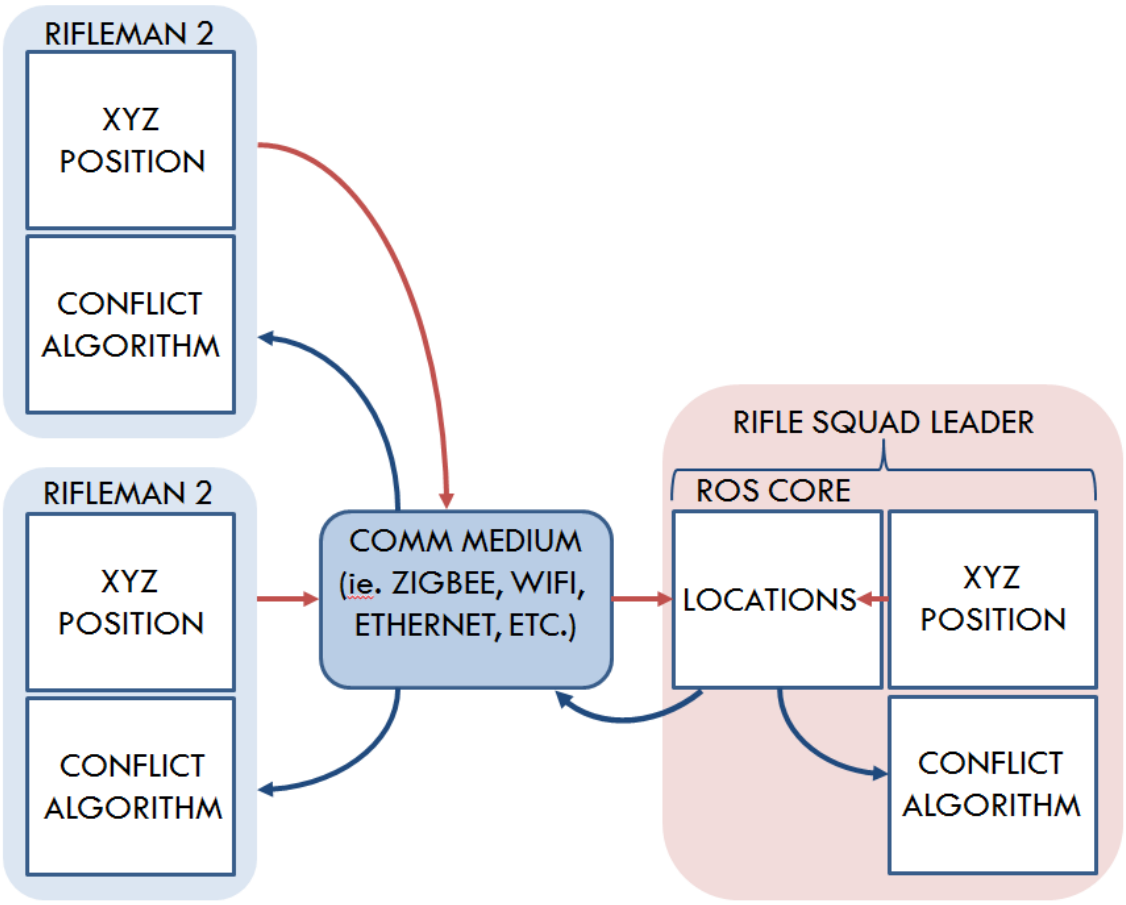


Figure 41. Data flow through the network goes from nodes to the core and from the core to the nodes.

With a concept of how the data flows through the network, a closer look is taken at the specifics of the code attached to the top of the program that handles the interactions between the program and the network. An excerpt from the conflict algorithm code is shown in Figure 42. The lines of code shown were added to the top of the conflict algorithm code to enable it to work with the network. The only other additions to the program were library import calls at the top of the code. The additional libraries are single lines of code and only reference ROS definitions. It is a requirement for the program to be able to access certain ROS features.

```
36 ####################    ROS    ##########################
37 # Define functions that will be called by the main script
38
39 def PosePub():
40         pub = rospy.Publisher('RifleData', String, queue_size = 20)
41         #'RifleData' will be the name of the node that broadcasts info about rifle
42         rospy.init_node('PosePub', anonymous=True)
43         #Inintializes the node under function call 'PosePub'
44         rate = rospy.Rate(5) #5hz
45         #It will refresh at 5hz
46         if not rospy.is_shutdown():
47                 PubOut = "Roll [%.2f] Pitch [%.2f] Yaw [%.2f]" % (roll, pitch, yaw)
48                 #Assembles variable with program variables
49                 rospy.loginfo(PubOut)
50                 #Logs the time stamp
51                 pub.publish(PubOut)
52                 #Publishes the data in variable 'PubOut'
53                 rate.sleep()
54                 #Pause after broadcast
55
56
57 def callback(data):
58         #This funciton retrives an x,y,z coordinate from ROS and...
59         #stores it in the local variable matrix posFriendly
60         global posFriendly
61         #Function will use program global variable
62         posFriendly[:,data.w]=matrix([[data.x],[data.y],[data.z]])
63         #Retrieve data from ROS message
64         return posFriendly
65
66 def listener():
67         #This funciton runs on a constant loop in a thread to constantly...
68         #listen to the ROS topic /NodePositions for a message type...
69         #Quaternion and when it retrives a message it calls 'callback' function above
70         rospy.Subscriber("NodePositions", Quaternion, callback)
71
```

Figure 42. The excerpt of code that integrates the program to the network.

There are two main divisions in this code, a section that sends data and a section that retrieves it. In this code, there is a portion dedicated to sending the roll, pitch, and yaw of the rifle. This data was not necessary to any other part of the network. The purpose of this code is to broadcast the data from the inertial sensors on the rifle to the data broker. It showcases the ability of the code to send and receive the positions as well as provided accuracy readings of the sensors for analysis. This code demonstrates the ability to not only deliver data to this program but also send data from this program to the network. That code is shown in lines 39–54. There are notes within the code that detail each line. The first line labels the function as ‘PosePub,’ the name that is called later in the code to run this section.

The second section of the code, lines 57–70, deals with retrieving the data from the network. Specifically, a loop is started as a thread that runs the function called ‘listener’ in line 66. The advantage of using a thread is that it runs as a separate process

within the program. As a separate process it is not dependent on the program looping back to run the line of code. It has a separate loop and process of its own. This function listens to a ROS topic called ‘NodePositions,’ as shown in line 70. Each time a new message is posted to that topic, it calls another function called ‘callback’ that is defined in lines 57–64. The function ‘callback’ takes the recently posted message from that topic and logs it into a matrix called ‘posFriendly’ that is used later to compute conflicting geometries.

## K. FINAL VALIDATION EXPERIMENT

This set of experiments accurately supplements the location element of the network. The Vicon system is used as an indoor GPS solution to determine the location of each node on the network. Vicon does this through Infrared cameras located throughout the room and infrared reflective beads attached to each object that is tracked. The cameras emit infrared light that reflects off of the beads and through several different camera angles, and the objects is able to be tracked three dimensionally throughout the room. That location data is posted to the ROS topic called ‘NodePositions.’ Each rifle connected to the network (in case one) is able to access the location of the other nodes and compute the conflicted geometry zones. Several experiments were conducted to test the capability and robustness of the network and the algorithms used to compute positions and firing geometries.

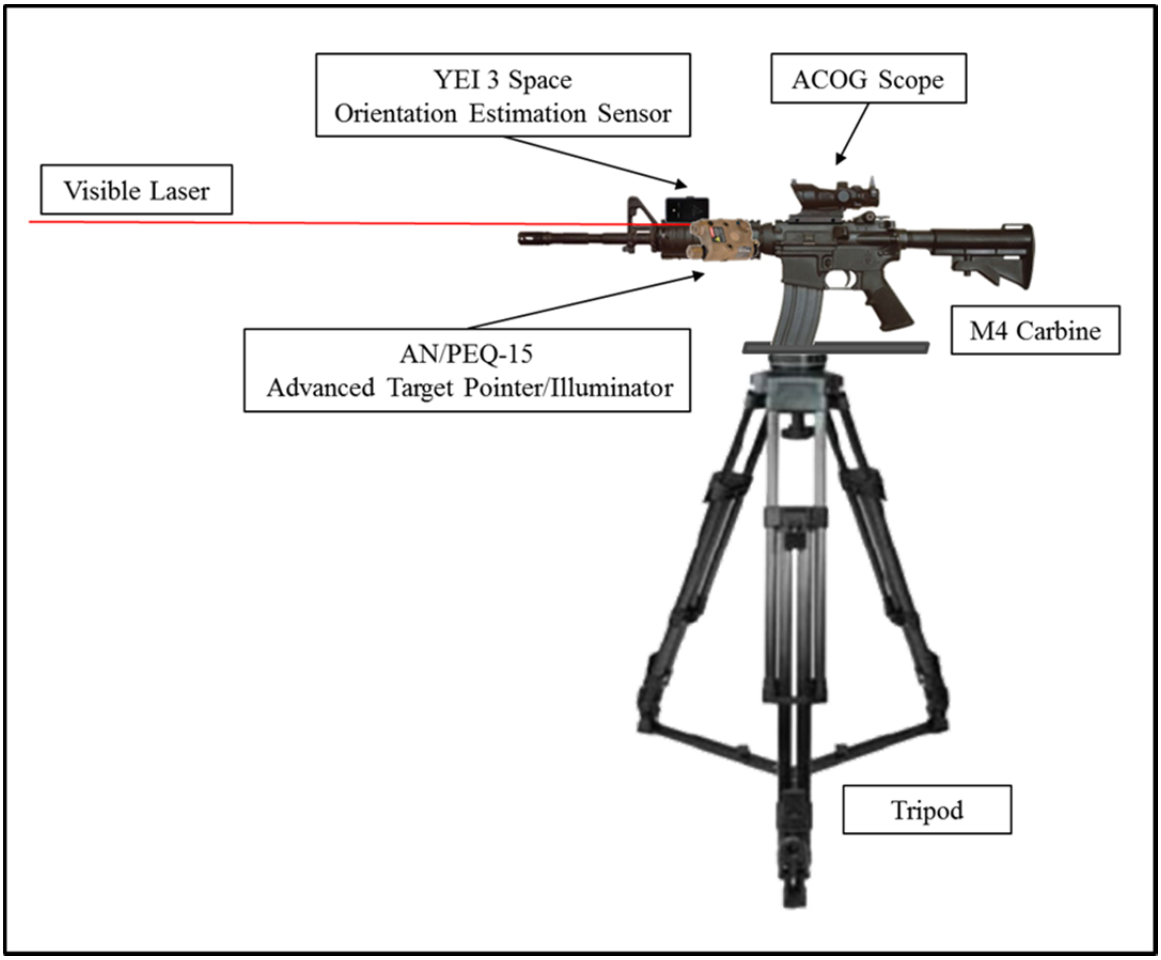


Figure 43. Example of how the rifle was set up for these experiments, from [26].

Experiment 1:

The rifle and all targets are stationary. The rifle is swung on the tripod to point at different nodes and test for cease fire conditions. The first test is with four nodes, then with ten, then twenty, which tests the algorithms ability to hold 20 positions. The positions in the NodePosition array do not update. It is expected that the array is filled with all positions in the programs first 20 cycles. Lag time for the mathematical calculations is assed. The only variable is the size of the position array.

Experiment 2:

The rifle is stationary. Some targets are fixed and others moving. This tests the networks ability to update node positions in the array and the ability of the rifle's node to pull down those positions and compute whether a conflict exists. The cease fire trigger should seem instantaneous to a human user. In other words, the reaction is less than a half second. The number of moving targets is limited by the number of people in the laboratory.

Experiment 3:

The rifle is moving, but all targets are stationary. This is similar to Experiment 1 in network testing but tests the algorithm's robustness with respect to its own position change. Performance is assessed accordingly.

Experiment 4:

Both rifle and targets are moving. This tests the computing power of the processor and the delivery capabilities of the network.

## L. CHAPTER SUMMARY

Many of the original goals of the experiment are achieved through this experimentation. The network is expandable to include new data types, functions, and abilities. It is able to deliver data to those who need it. New features can be added to the existing hardware platforms, especially the Raspberry Pi. The network transmits and receives over several transmission mediums simultaneously. In addition to the different communication protocols, there are several different hardware platforms, coding languages, and sensors all working together in harmony. The network has proven the ability to have interoperability through different off the shelf platforms. There are currently no levels of encryption in the pathways, but that can be added by adding encryption to the infrastructure, such as a Wi-Fi encryption protocol.

As a proof-of-concept, the fratricide detection program was employed on the network. The network successfully transmits the location of each single node in such a way that the data gets to those who need it. Those nodes requesting the positions of the other nodes then calculate if the direction of their field-of-fire conflicts with the position

of any friendly units. If that mathematic manipulation determines a conflicting field-of-fire, the warning displays to indicate an unsafe condition.

Pathways have been built into the network to incorporate more topics and more data throughput. Not only is the network able to distribute the data amongst its subordinate nodes, but the data dispersed throughout the network is also accessible by exterior authorized sources. Another computer can reach in and view the topic of the node positions and then use that data to plot a graphical output of each individual squad member in real time. When that information is integrated with a Google Maps overhead satellite image, more of the tactical picture comes into view. Greater situational awareness is achieved through better communication.

There are currently more projects that seek to add more applications and abilities to the network. Several are focused on the increase of functionality in the Marine carried device—the Raspberry Pi in our experiments. Others still are focused on a larger network integration. This would enable the ability of this small tactical MANET to integrate with a larger network to take advantages of resources found on Department of Defense networks and the Internet. Not only does the data flow form the MANET up to the higher networks, but now the higher networks are able to provide resources and information to the smaller networks.

# V. RESULTS

> Thus, it has come about that our theoretical and critical literature, instead of giving plain, straightforward arguments in which the author at least always knows what he is saying and the reader what he is reading, is crammed with jargon, ending at obscure crossroads where the author loses its readers. Sometimes these books are even worse: they are just hollow shells. The author himself no longer knows just what he is thinking and soothes himself with obscure ideas which would not satisfy him if expressed in plain speech.
>
> —Carl Von Clausewitz

The work accomplished in this thesis research shows that a dynamic data driven network is possible and, in fact, easily implemented into a MANET. A mobile deployable network capable of being carried by dismounted personnel was prototyped and tested. Goals established by the tenets of network-centric warfare were met, and a response to the call for squad based technologies has begun to be answered. This foundation paves the way for other technologies to further utilize the network and enable the dismounted squad. This network is an applicable next step to technologies like encrypted radios, Blue Force Tracker, and GPS. Just as the construction of communication pathways lead to and built the Internet, so this network further brings together those who use it and enable further technological growth.

## A. RESULTS OF PROTOTYPE

As the research suggested, a Data Broker Scheme is deployable with a MANET. This work is the first—to our knowledge—of these schemes to be prototyped and tested. As the first prototype of this system, it proves the concepts, theories, and simulations of the research before it. A working prototype of the network, implementing ROS as the data broker, now works and is able to interconnect multiple mobile platforms. The low overhead of the network means that it can run on small mobile devices such as Raspberry Pi and Arduino. Additionally, these small devices allow for a light-weight and portable component for a squad of infantrymen to carry.

The network successfully demonstrated its ability to interconnect these devices through an experiment employing a friendly-fire detection program across the network. Each device was able to effectively communicate with one another and enabled the operation of the program. By nature of the friendly-fire application, it was reliant on the delivery of information from external sources. This prototype network was able to deliver that information through a data broker managed MANET.

As this project stands now, it is an operating data broker MANET transmitting the data necessary for applications to run on the network. Currently, it distributes unit location and pose computations for conflicting fire-zone analysis. That functionality includes the sensor package attached to the Raspberry Pi that interfaces with the greater network. In real time the pose and position is calculated to determine conflicting geometries with other nodes of the network, whether those nodes are other riflemen carrying the same sensor package or simply position markers. There are no rifles associated with the position markers. The ability to accurately determine location was an assumption of this research, and as a result, the position data was provided from an external source.

ROS has been uniquely implemented as this MANET's data broker. The data flowing through the network is stored in the ROS topics on a main node that is running the lightweight. From any other node on the network, those topics are visible and accessible. Those topics can also be posted to, altered, or used by any node on the network. In this project, the topics were monitored for purposes of visually confirming dataflow and for confirmation of manually injected positions to the network.

## B. GOALS ACCOMPLISHED

The project began with several goals, both theoretical and technical. The driving philosophies came predominately from the network-centric warfare doctrine and the technical goal of delivering technologies to the infantry squad.

### 1. Theoretical Goal

The ability to communicate is the oil in the machine of strategy. Unified action, efficiency, safety, and even victory rely on clear, concise and effective communication. Network-centric warfare calls for an enhancement of this communications network in order to bring shared situational awareness to the battlefield and further provide for effective decision making. Progress towards these tenets has been greatly advanced through this work; however, this progress is only the beginning of further advances that must made as more technologies utilize this network.

Through the network developed for this project, the groundwork has been laid for a system that makes more information available to the troops on the ground. This network enables systems that can ensure troop safety and effectiveness. The goal is to ease the burden of uncertainty and enhance their situational awareness. With more information, they are able to maintain a dominant edge over the enemy. This makes their lives easier, their work more effective, and works to make them safer.

### 2. Technical Goals

Network-centric warfare calls specifically for technological advances in military equipment to enhance the abilities of our troops. This work physically provides a technological advancement for just that purpose. It is far from field-ready, but it is the first build prototype of its kind.

Additionally, this network serves as a partial answer to a networked infantry solution. The goal of bringing wearable, portable electronics to the infantry squad requires the foundation of a communications network. This network can serve as a solution to that problem. There are many available choices for network architectures and physical communication protocols, but this version shows potential to be successful for small unit operations.

## C. TECHNICAL IMPROVEMENTS

### 1. Data Dissemination

Data accessibility through enhanced communication in a mobile ad hoc network is a key take-away from this network. Data can be accessed at any node, at any time, without rearranging nodes, connections, or reprogramming other components. Communication has improved the availability of data and ensured its delivery to the right people at the right time. The ability to distribute data through the network achieves the research goals of the research. The goals outlined by network-centric warfare to include situational awareness have been improved, and decision making is better informed and, therefore, better equipped.

A node can come into the network and leave the network without affecting dataflow. This is unique from a network perspective because the typical system cannot function correctly if parts of the network are missing. Each node is unaware of any other nodes on the network—only the topics. Still, the fact that the network can be incrementally initiated is a leap ahead of a traditional co-dependent network. Additionally, because of the open accessibility to the data passing around the network, certain parts of the network can be brought on and offline as the need presents itself. For example, with the fratricide detection program, the network passes position data to the other nodes on the network. If desired, a software node can be launched that uses a graphical layout to map the positions of each node and track their movement in real time. Another example is a video feed being posted through the network. With the raw data feed, it can be recorded and stored on a server, passed through a facial recognition program, recorded and mapped into a training video, streamed lived to other locations, and more. All of these implementations can be brought in and out of the network without changing or altering the node that is broadcasting the video feed. This network brings a new dimension to scalable, robust, and multifunctional data. The data is free to flow to those who need it.

Nodes have more versatility in this network. Traditionally, an element of the network must know where to transmit data. It must have an endpoint address to send its

data. In this build, the data stream is left open-ended. The topic is labeled in a variable name instead of an IP address and needs no endpoint other than that. No other node needs to begin using it; no other processes must be altered. The only element of coding used to initiate that topic originates in the node itself. Other nodes that use the data must know the variable name of that topic to find it, but that listing is public to the network users. A node seeking information from a publishing node does not crash if the topic is not visible, it simply holds until it is online. Though not ideal, a pause in processing is more desirable in a connection break than having to restart the device.

### 2. Improvements from Other Networks

This network is an improvement over the two typical data dissemination networks: point-to-point relay and open channel communications. Instead of sending point-to-point communications and having to relay data to each party, this network allows the data to be openly available to all parties. The relay can be thought of as a multicast data protocol. Additionally, it avoids the party-line drawbacks of an open channel communications scheme. When a party-line is used, the data is sent to every user, whether they want it or not. This quickly causes information overload for human beings and slows down the processor of a low-resource device. The amount of data can certainly be handled but not in every circumstance. This improved functionality over other networks topologies is largely due to the use of the Managed Data Broker scheme. The ability to deliver information when and where it is needed is achieved by the data management abilities of the data broker.

Traditionally, there are two other ways of implementing such a network without a data broker scheme. The first and most common is a variation on cloud services. A node would send its data to a server running a PHP and SQL database to catalog positions. Next, a separate program would recover that data from the database. This implementation is ideal for large operations in which the data needs to be stored permanently. In fact, a database similar to this is how data would be stored long term in the network built during this work. The difference is in setup and implementation. Traditionally, the PHP and SQL database needs to be established not only in software but also physically. In the network

built here, the caching of data is stored locally for redistribution to the nodes. There is less overhead for the redistribution of the data. This allows for tactical, quickly-deployable networks. This network can also be built over several mediums and network protocols including serial connections, Bluetooth, CAT5 cable, Xbee, Wi-Fi, and USB connections.

### 3. Interoperability

With the ability to operate the network over several different devices and share data between programs written in different languages comes great flexibility. There is great interest in the ability for future military systems to have the ability to operate with each other. Currently, many of the systems the military employs utilize proprietary communications protocols. With this network, the ability exists to operate across these multiple platforms. This is especially important as more military systems become autonomous. The integration of ROS to robotic autonomous systems can be easily implemented into the network. Through that interoperability, control of these autonomous systems can be routed through the network for distributed control of resources.

### 4. Weaknesses

Currently, there are possibilities for several cyber vulnerabilities to the system. Most of the operations of the ROS network are not encrypted and, because they are open source, would be easy to infiltrate. There are some precautions built into the wireless transmissions. With the Xbees, the baud rate, parity, bitrate, and frequency are unique to the system, but those can be guessed over time. In addition, the data is simply coming across as a bit stream. That data can be encrypted through the Xbee protocols. The same is true with the current CAT5 cables and Wi-Fi signals. These are not encrypted or locked, however, certain ports can be locked and kept secret. Encryption and passwords can be added to the network.

A second fault of the system is that ROS does not have full support for various platforms. The build used on the Raspberry Pi is a partial build that does not have complete functionality. Luckily, it had every feature that was necessary for this network. The builds for platforms like the Gumstix may not be as complete. With an open source

program like this, there is no dedicated support from a company to keep it working, but there is the ability of skilled programmers to rewrite parts of the code. A more skilled programmer could augment a version of ROS to work on different systems. That is exactly how the version used on the Raspberry Pi was created. The question is whether to use ROS as a continued operating system from the network or to build a new one that is more secure. Advantages of keeping ROS are ease of integrating the network into robotic systems. A disadvantage is the open availability of the code.

### 5. A Buildable Product

Perhaps the most distinct improvement made through the prototype developed for this thesis is the ability to rapidly augment existing components into the network. Many systems that are in development or production can be easily integrated with this project. ROS is built from the tools and languages that every developer starts with: Java, C++, Python, and more. There are literally thousands if not hundreds of thousands, of projects, platforms, sensors, actuators, drones, robotics, analysis software, displays, and more that are only lines-of-code away from integrating to the network.

The basic concept of this simple integration is depicted in Figure 44. Given a simple program, only a small amount of code needs to be added to integrate it to the ROS network. As seen in Chapter IV, the code is not very complicated. It involves the mapping of local variables of the program to the message types used by ROS. That message is then assigned to a topic within the network. With that type of simplicity, nearly any program can be added to this network, including software that operates sensors and actuators as well as software that simply analyzes data.

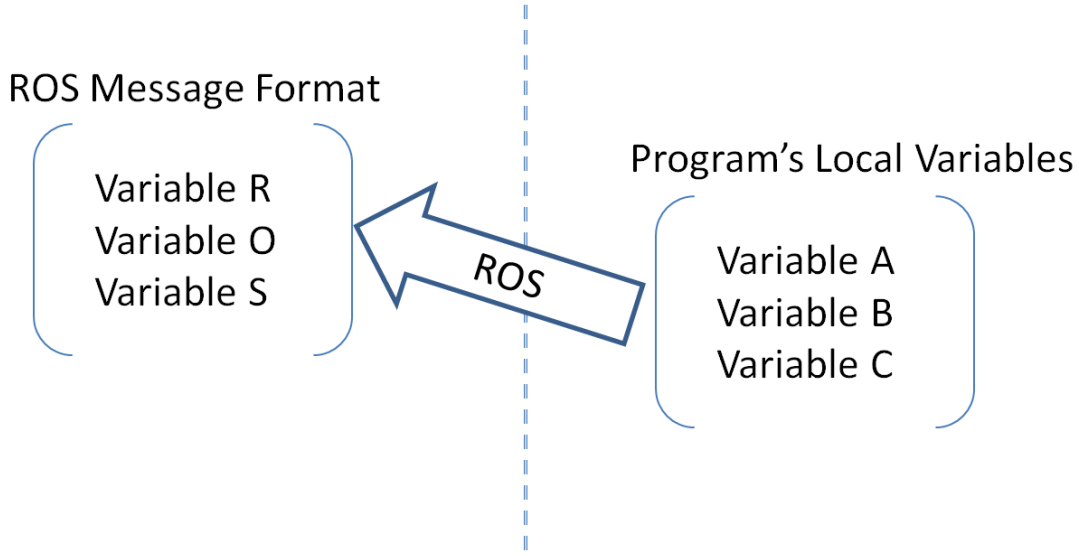


Figure 44. Illustration of how local program variables are duplicated into a ROS message for the network.

## D. CHAPTER SUMMARY

This working prototype completed many of the goals set out for this research. The improvements to communication deliver greater ability to the ground units and shared contextual situational awareness through data dissemination. Decision making is more informed through ground troops' situation data. The time necessary for the communications between the parties is greatly reduced since they are receiving the same information displayed in the same manner. Tenets 1 and 2 of the network-centric warfare goals are reached through shared data and enhanced situational awareness. Finally, through new technology, the Marines themselves can be safer and more effective. Their carrying weight has been minimally increased through the addition of few components.

Though the network completed the goal of low cost data dissemination, it is far from ready for the field. The prototype built here serves the purpose as an academic test bench for rapidly prototyping functionality. There must be improvement to the security of the communications as well as material improvements to the hardware and power supplies. There is much more functionally that can also be added to the system to improve its usability and capability. Those future abilities are discussed in the next chapter.

# VI. FURTHER WORK

> There are very few men-and they are the exceptions-who are able to think and feel beyond the present moment.
>
> —Carl Von Clausewitz

This network is simply a foundation upon which further technologies will be built. The design of this network and the choices made in the implementation were intended to lead to the easy implementation of existing technologies to the network as well as lay ground for simple implementation of future ideas. As the network stands now, the framework has been initiated. The response has been made to the call for integrated systems, adaptable networks, and tactical integration. This implementation is not the final product, but it is a robust, adaptable bench-top model available for prototyping and testing. A system such as this is an open sandbox to inventors and researchers.

## A. FUTURE ABILITIES

This network is an enabler of future technology. There are several abilities that can be added to the squad using the network. If the full potential of the network were to be utilized, greater data dissemination would cause vast improvements to the operational capabilities of dismounted infantry. Many of these ideas stem from the requests of DARPA in the Squad X Initiative. Every element brings new data to the network, which in turn opens the ability of the network to analyze and react to that data. These ideas include but are not limited to:

- Health and vital monitoring of individual units, as well as analysis and response programming to changes in these values.
- Weapons release sensing and ammo depletion sensing.
- Programmed drone response to low ammo supplies and medivacs.
- Speed dial like support requesting. “Press 1 for bulldozer.”
- Alternative energy production solutions to power the system.
- Encryption and electromagnetic spectrum protection to include frequencies with high bandwidth and low probability of intercept or detection signals analysis.
- Camera system with remote video feed.

- Facial recognition of camera system comparing with remote linked facial database
- Modeling software that creates overhead display of units.
- Indoor GPS solution.
- Software and coding interface for this network to BFT.
- Form factor design for components to be carried by user.
- A library of ROS functions and messages tailored for these applications.
- Integration of robotic and autonomous systems to the network.
- Integration of sensor network to the system.
- Development of disposable, deployable sensors for the network.
- Integrating of data pulled from the Internet such as satellite overlay, weather, etc.
- Heads up display with products like Google Glass.
- Wearable electronics integration with products like smart watches, phones, and tablets.
- Propagation and broadcast studies to determine antenna design and broadcast methods.

The Army has some ideas about the technological abilities it hopes to see in the future. In a brief article about these ideas, they mentioned the desire to bring technological improvements to the field. A concept of the abilities they see coming in the future of warfare are depicted in Figures 45 and 46.

Lincoln Labs has several ideas similar to those listed above that could be implemented, shown in Figure 47. Their design uses an FPGA chip to run the nodes rather than a Raspberry Pi. They include sensors for audio and visual inputs as well as others that monitor the individual's vitals. There are tablets used as interfaces for the user and the network. These components could also be implemented into the ROS network created through this research.

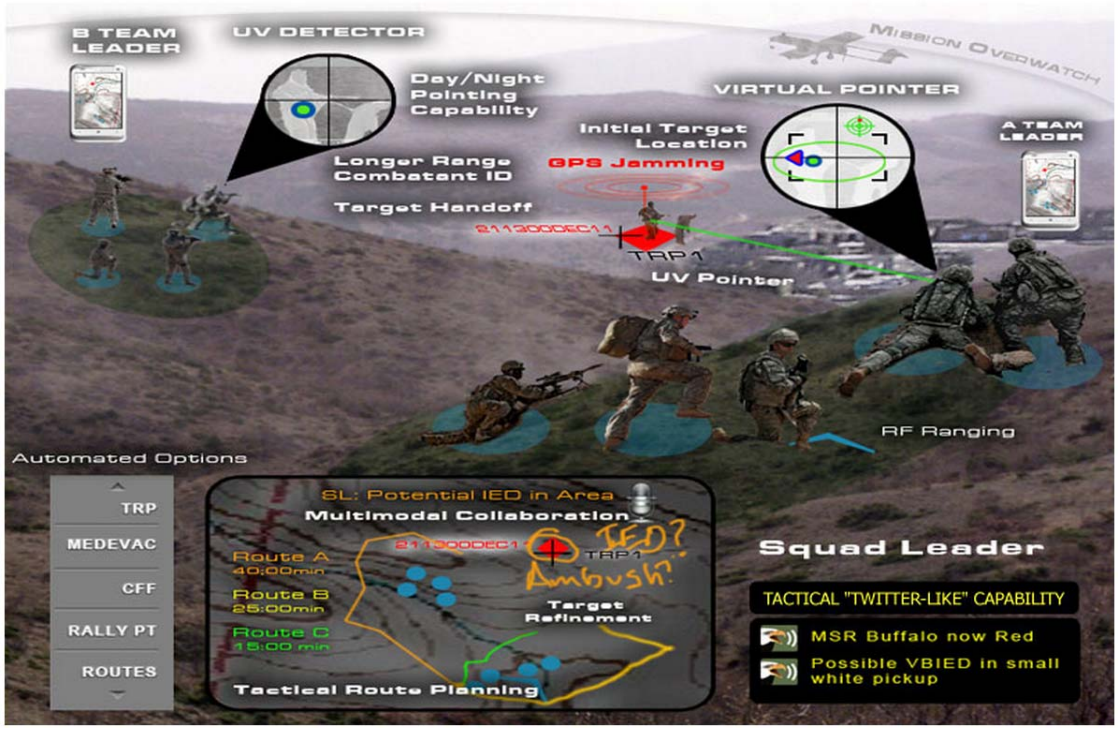


Figure 45. Example of the capability advantage possible if the right networking capabilities are delivered, from [27].

| Proactive Decision Support and Collaboration | Increased SA/SU | Faster and More Accurate Target ID and Handoff |
|---|---|---|
| • Proactive Alerts<br>• Ops/Intel Information Collection<br>• Tactical Twitter: OPS/Intel Notification<br>• Tactical Automated Options: Points of Interest E.G., TRPS, RPS, HLZS, ETC. | • POS/NAV in GPS-Denied Environment<br>• Leader's Dashboard<br>• Dynamic AAR<br>• Cross CE Data Exchange | • Virtual Pointer<br>• UV Pointer and Dectector<br>• Active SWIR<br>• Leader Effects<br>• Management |

Figure 46. More examples of the capabilities that can be developed for the network, from [27].

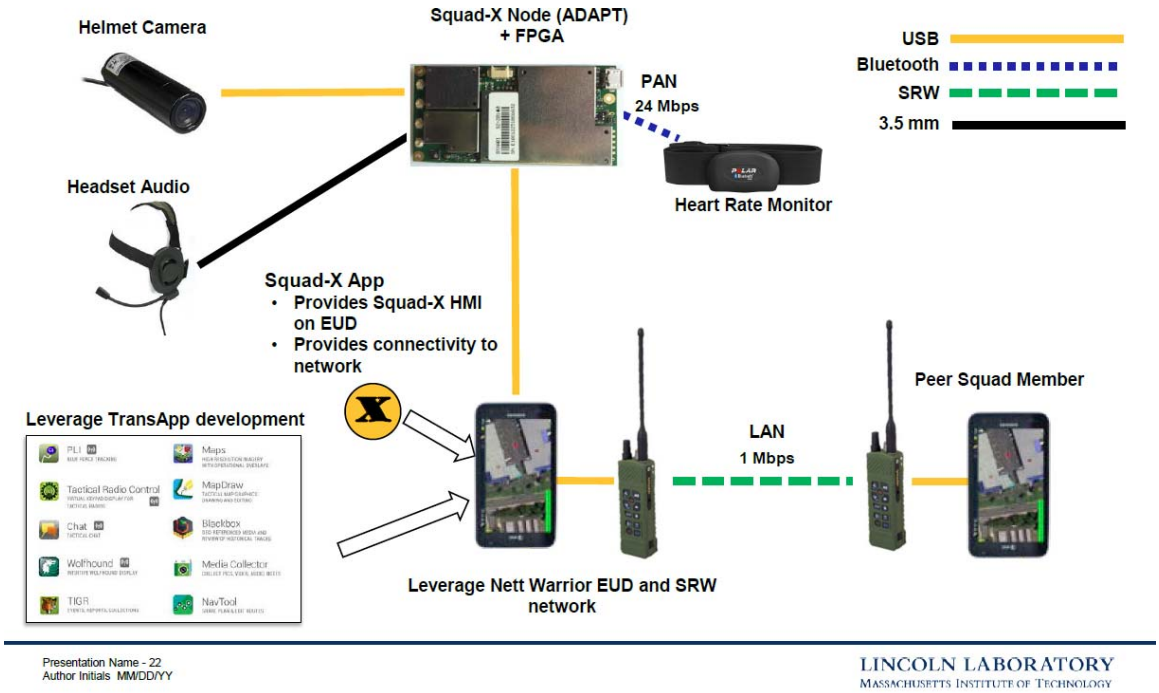


Figure 47. Lincoln Labs concept of future technology integration, from [13].

## B. FUTURE INTEGRATION

Specifically, the integration of this network into a larger network will be a key improvement. If a piece of software can be implemented that interfaced the data broker with a data base server, that will demonstrate the cross-structural integration ability of the system. The data broker will have to be able to request and search for information in that database in such a way as to specifically obtain what is needed. This is an interesting problem because the data broker will have to subscribe to a node that is not actively publishing and have to request a publication stream from a stagnant source. As the structure stands now, there is the ability to initiate an "as needed" publication, but the conditions need to be programed.

Integration to a larger network that may or may not be data broker managed is the next step in bridging the small-scale network discussed in this thesis research to the large-

scale networks currently used by the military. The potential network connections that can be made that will integrate the network discussed here into the larger network-centric warfare setting is shown in Figure 48, which is an example of how the friendly-fire scenario can benefit from this large-scale connection. Specifically depicted is the ability to relay its information to leaders in Washington as well as receive important information from them such as blueprints of the building, images of high value targets, or details about aerial support.

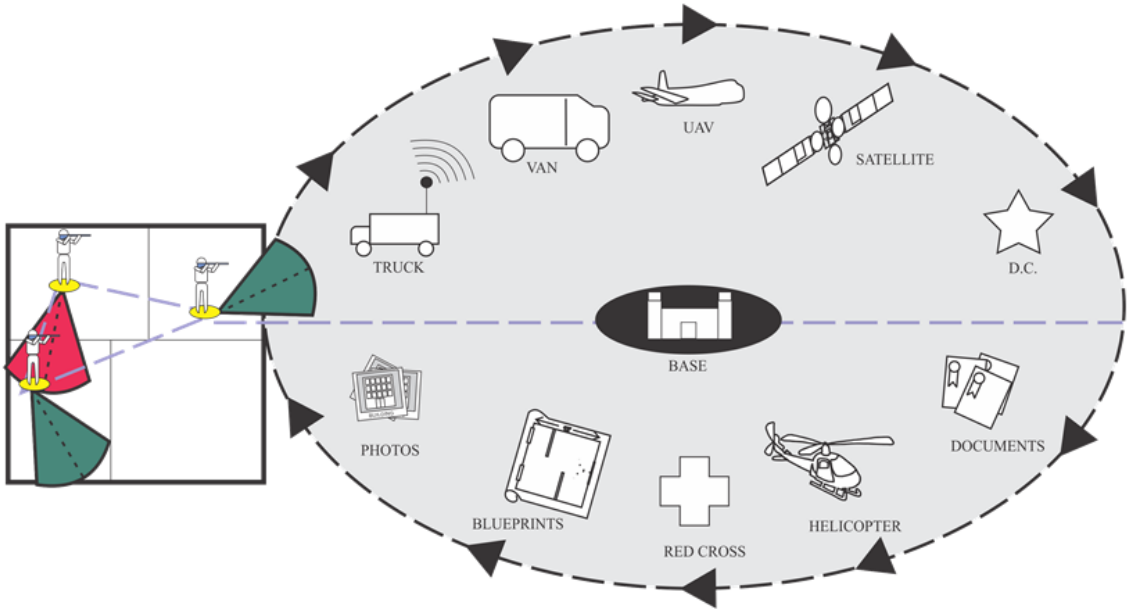


Figure 48. Example of how the network can connect into a larger network.

Perhaps in the future the most significant ability of the network will be built-in functionality and interoperability with robotic and autonomous systems. As the government and military move to automate and subsidize dangerous jobs with autonomous systems, they need to integrate the network and provide information and situational awareness rapidly. Many of those robotic systems are being developed using ROS as an operating system. The future integration of these systems will be much easier because they already possess the ROS interface.

## C. TRAINING APPLICATIONS

Currently, the military uses several methods to train and prepare on a tactical level. Much of the practice is done with either paintball modified rifles or laser modified rifles similar to laser tag. These systems, however, are physically different from the equipment the units actually use. The network developed for this project can be used on the *actual* rifles used in combat, eliminating the cost from three sets of rifles to one. More importantly, this allows them to train with the equipment they will actually use, and creates more realistic training environments for our military.

Commanders can simulate, record, replay, and analyze training situations. Through replays of the data, they can critique rifle position, speed of movement, speed of rifle sweep, conflicting fire zones, physical position on the battlefield, response time of units and more. Just as a sports team watches tapes of their previous games, a group using and recording the right data with this network has a tactical advantage.

## D. CHAPTER SUMMARY

There is an unlimited amount of functionality and capability that can be enabled through this network. Think of the growth in ability in mobile apps as the hardware on smart phones improved. Fingerprint scanners, heart rate monitors, GPS tracking, and more have been added to a smart phone. Imagine if the same efforts were made on this network. The more sensors that are added to the node, the greater the types of data can be sent across the network. With greater amounts of data come greater ways to integrate, analyze, and use that data.

Perhaps the greatest contribution to this network is the software that runs on the network and analyzes the data. Just as Google has used large data to predict and assist people, the analysis of this data can warn Marines of inclement weather, suggest alternate routes, and route important traffic through the network. Being able to analyze and predict the needs of the Marines is essential to the future battlefield. Perhaps, eventually this technology will be used to analyze and predict the enemy’s movements and suggest strategies to defeat him.

There are applications for this product outside the military as well, with emergency services and disaster relief organizations. Humanitarian aid and disaster relief have been a significant focus in the news and government over the past decade, and as this technology is applied to this area, perhaps it will help. There are also applications in sports in being able to record the actions and motions of athletes individually and as a team. The recorded data will allow for improvements and analysis and perhaps even warn about imminent injuries. The possibilities are great.

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF REFERENCES


[1] S. Erwin. (2011, Feb. 23). U.S. troops loaded with technology, but can't harness the Power of the network. *National Defense Magazine* [Online]. Available*:* http://www.nationaldefensemagazine.org/blog/Lists/Posts/Post.aspx?ID=327

[2] S. Erwin (2010, Sep. 1) Army under Pressure to bring broadband to the battlefield. *National Defense Magazine* [Online]. Available: http://www.nationaldefensemagazine.org/archive/2010/September/Pages/ArmyUnderPressuretoBringBroadbandtoTheBattlefield.aspx

[3] D. S. Alberts and J. Garstka, "Network centric warfare Department of Defense report to Congress," Department of Defense, Washington, DC, 27 July 2001. [Online]. Available: http://www.dodccrp.org/files/ncw_report/report/ncw_cover.html

[4] D. S. Alberts. (1996) *Information Age Transformation: Getting to a 21st Century Military* [Ebary version]. Available: http://www.dodccrp.org/files/Alberts_IAT.pdf

[5] D. Dusseau, J. A. Negro, and B. Clinton, "Designing user friendly situational awareness products," presented at DASC 20th Conference, Daytona Beach, FL, 2001.

[6] C. Krulak. (1997) *Warfighting*, *MCDP1* [Ebary version]. [Online]. Available: http://www.clausewitz.com/readings/mcdp1.pdf

[7] D. Stephens, C. Magsombol, and N. Browne, "Network programming of Joint Tactical Radio System radios," presented at MILCOM, San Diego, CA, 2008.

[8] R. Dunn, "Blue Force Tracking: The Afghanistan and Iraq experience and its implications for the U.S. Army," Northrop Grumman Mission Systems, Reston, VA, 2003. Available: http://www.northropgrumman.com/AboutUs/AnalysisCenter/Documents/pdfs/BFT-Afghanistan-and-Iraq-Exper.pdf

[9] K. Hall. (2008, Jan. 12) Friend, or foe? Blue Force Tracker to clear fog of war for Next Afghan PRTs. [Online]. Available: http://www.laughlin.af.mil/news/story.asp?id=123082048

[10] "Understanding voice and data link networking," Northrop Grumman, San Diego, CA, Dec. 2013. Available: http://www.northropgrumman.com/Capabilities/DataLinkProcessingAndManagement/Documents/Understanding_Voice+Data_Link_Networking.pdf

[11] J. Conatser, “Force XXI Battle Command Brigade and below-Blue Force Tracking (FBCB2-BFT). A case study in the accelerated acquisition of a digital command and control system during Operations Enduring Freedom and Iraqi Freedom,” MS. thesis, School of Business, Naval Postgraduate School, Monterey, CA, 2005.

[12] “Squad X core technologies (SXCT),” DARPA Tactical Technology Office (TTO), Feb. 2015. Available: https://www.fbo.gov/index?s=opportunity&mode=form&id=9ebb7959887c09e64c47ad7ab31c638b&tab=core&tabmode=list&=

[13] J. Muldavin, “Squad-X architecture study,” Lincoln Laboratory Massachusetts Institute of Technology, Dec. 2014.

[14] “Marine inertial navigation system data sheet,” Raytheon, Anschutz, Germany. 2015. Available: http://www.raytheon-anschuetz.com/fileadmin/content /Downloads/Brochures/marine-inertial-navigation-system-2.pdf

[15] T. Gadeke et al.. (2012, October 3). Smartphone pedestrian navigation by foot-IMU sensor fusion. *Ubiquitous Positioning, Indoor Navigation, and Location Based Service*, pp. 1–8. Available: 10.1109/UPINLBS.2012.6409787

[16] J. Rantakokko et al.n. (2011 April 15). Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization. *IEEE Wireless Communications,* 18(2). pp. 10–18. doi: 10.1109/MWC.2011.5751291

[17] S. Y. Oh, D. Lau and M. Gerla. (2010 October 20). Content centric networking in tactical and emergency MANETs. *Wireless Days.* pp. 1–5. doi: 10.1109/WD.2010.5657708

[18] Broker vs. Brokerless. (2012 May 24). Zero MQ. [Online]. Available: http://zeromq.org/whitepapers:brokerless.

[19] I. Lahyani, W. Makki, and C. Chassot, “Failure prediction for publish/subscribe system on MANET,” in *IEEE 21st International Workshop Enabling Technologies: Infrastructure for Collaborative Enterprises,* Toulouse, France, 2012. pp. 98–100. doi: 10.1109/WETICE.2012.18

[20] J. Haillot and F. Guidec, “Towards a usenet-like discussion system for users of disconnected MANETs,” in *Advanced Information Networking and Applications—Workshops*, Okinawa, Japan, 2008, pp. 1678–1683. doi: 10.1109/WAINA.2008.71

[21] M. Pandey and B. D. Chaudhary, “A reconfigurable distributed broker infrastructure for publish subscribe based MANET,” in *Sensor Networks, Ubiquitous and Trustworthy Computing*, Taichung, 2008, pp. 361–366. doi: 10.1109/SUTC.2008.30

[22] N. Jianwei, L. Chang, X. Yongping, and M. Jian, “A publish-subscribe algorithm for environment awareness in MANETs,” in *IET International Communication Conference Wireless Mobile and Computing*, Shanghai, China, 2009, pp. 681–685.

[23] Arduino’s Product Page (n.d.). Arduino. [Online]. Available: http://www.arduino.cc/en/Main/ArduinoBoardUno. Accessed June 1, 2015.

[24] Raspberry Pi website (n.d.). RaspberryPi. [Online]. Available: https://www.raspberrypi.org/raspberry-pi-2-on-sale/. Accessed June 1, 2015.

[25] Robotic Operating System (ROS) website (n.d.). ROS. [Online]. Available: http://www.ros.org/. Accessed June 1, 2015.

[26] C. Khan, “[Currently Untitled],” M.S. thesis, Dept. Electron. Eng., Naval Postgraduate School, Monterey, CA, 2015. [Currently not published]

[27] Mission command/actionable intelligence technology enabled capability demonstration. U.S. Army Communications Research, Development and Engineering Center. [Online]. Available: http://www.cerdec.army.mil/inside_cerdec/core_technology/mission_command/mission_command_and_actionable_intelligence_TECDs/. Accessed June 1, 2015.

THIS PAGE INTENTIONALLY LEFT BLANK

# INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California